

Muhammad al-Xorazmiy nomidagi Toshkent
Axborot
Texnologiyalari Universiteti

Dasturlash 2

C++

USLUBIY QO'LLANMA

Oybek	Mallayev
Zamira	Abdullayeva
Yulduz	Baxramova

**O‘ZBEKISTON RESPUBLIKASI INFORMATSION TEXNOLOGIYALARI
VA KOMMUNIKATSIYALARINI RIVOJLANTIRISH VAZIRLIGI**

**MUHAMMAD AL-XORAZMIY NOMIDAGI TOSHKENT AXBOROT
TEXNOLOGIYALARI UNIVERSITETI**

KOMPYUTER INJINIRINGI FAKULTETI

«Informatika asoslari» kafedrası

Mallayev O.U., Abdullayeva Z,Sh., Baxramova Yu.Sh.

«Dasturlash 2» fanidan laboratoriya ishlarini bajarish uchun uslubiy qo‘llanma

Toshkent - 2020

Mualliflar: Mallayev O.U., Abdullayeva Z,Sh., Baxramova Yu.Sh.

Kunduzgi yo‘nalish talabalari uchun “**Dasturlash 2**” fanidan laboratoriya ishlarini bajarish uchun uslubiy qo‘llanma. TATU. 245 b. Toshkent 2020.

Birinchi semestrda talabalar algoritm va C++ dasturlash tili bo‘yicha umumiy bilimlarga ega bo‘ladi. Ikkinchi semestrda **OYD, Collections, fayllar bilan ishlash, GUI** texnologiyalari yordamida visual ko‘rinishga ega, obyektga yo‘naltirilgan dasturlash yondoshuvlari asosida ishlovchi amaliy dasturlar yaratish ko‘nikmalariga ega bo‘ladilar.

Qo‘llanmadagi har-bir laboratoriya ishlarini bajarish algoritmlari va dasturlari visual ko‘rinishda, foydalanuvchiga tushunarli va aniq qilib yoritilgan. Mualliflarning fikricha talabaning o‘zi, mustaqil ravishda uslubiy ko‘rsatma yordamida kerakli mavzuni o‘zlashtira oladi.

Ushbu uslubiy qo‘llanma 2020 yil 23 iyunda Muhammad Al-Xorazmiy nomidagi Toshkent axborot texnologiyalari universiteti Ilmiy-uslubiy kengashining majlisida ko‘rib chiqildi va 9(134) - sonli buyruq bo‘yicha nashr etishga ruxsat berilgan.

Muhammad al-Xorazmiy nomidagi Toshkent axborot texnologiyalari
universiteti, 2020

KIRISH

Uslubiy qoʻllanmada obyektga yoʻnaltirilgan dasturlash tamoyillari asosida kichik loyihalar yaratish usullari keltirilgan. Jumladan kichik loyiha yaratish uchun turli xil kompilyatorlarda visual dastur tuzish mahoratini shakillantirish. Visual dasturlashning tarkibiy qismlari: nomaʼlum tiplar va nomaʼlum nomlar fazosi, konteynerlar, assotsiativ va tartiblanmagan konteynerlar, standart algoritmlar va iteratorlar, sonli sinflar bilan ishlash, sintaksis tahlil, murakkab saralash algoritmlari, amaliy dasturlash, komponentalar bilan ishlash, muloqot oynalari bilan ishlash, OLE, MFC texnologiyalari, koʻp oynali muhitlar bilan ishlash, kichik loyihalarni yaratish, foydalanuvchi interfeysini loyihalash va zamonaviy dasturlash muhitlari bilan integratsiya amaliy misollar koʻrinishida keltirilgan.

Uslubiy qoʻllanma obyektga yoʻnaltirilgan dasturlash tamoyillari, Visual C++ dasturlash muhiti komponentalari hamda ularning xossa va hodisalarini oʻrgatish asnosida visual koʻrinishga ega boʻlgan dasturiy vositalarni yaratishga hamda kichik loyihalarni ishlab chiqish koʻnikmalarini shakillantirishga bagʻishlangan.

Uslubiy qoʻllanma 5330200 - Informatika va axborot texnologiyalari, 5330500- Kompyuter injiniringi, 5330600 - Dasturiy injiniringi, 5350100 - Telekommunikatsiya texnologiyalari, 5350200 - Televizion texnologiyalari, 5350300 - Axborot-kommunikatsiya texnologiyalari sohasida iqtisodiyot va menejment, 5350400 - Axborot-kommunikatsiya texnologiyalari sohasida kasb taʼlimi, 5350500 - Pochta aloqa texnologiyasi, 5350600 - Axborotlashtirish va kutubxonashunoslik va 5330300 - Axborot xavfsizligi taʼlim yoʻnalishlarida tahsil olayotgan talabalar hamda mazkur sohaga aloqador professor-oʻqituvchilar, ilmiy xodimlar, magistrlar va mustaqil oʻrganuvchilar uchun moʻljallangan.

1- LABORATORIYA ISHI

1. Noma'lum nomlar fazosi va tiplarni dinamik tarzda aniqlash

1.1. Noma'lum turlar va noma'lum nomlar fazosi

Ishning maqsadi: C++ dasturlash tilida obyektga yo'naltirilgan dasturlash tamoyilaridan foydalanib, noma'lum nomlar fazosini tashkil qilish ko'nikmalarini shakllantirish.

Masalaning qo'yilishi:

Variantda berilgan class, konstruktor va class maydonlari yordamida noma'lum nomlar fazosini yarating. Class da obyektlar massivi ustida quyidagi funksiyalarni bajaruvchi 3 ta static metod yozing:

- **Qidirish metodi**
- **Saralash metodi**
- **O'zgartirish metodi**

Yuqoridagi metodlar uchun class maydonlaridan ixtiyoriysi tanlansin.

Masala: *“talaba”* nomli nomlar fazosi va unda **“Talaba”** classini yarating. Class maydonlari: *Familiyasi, ismi, kursi, baholari, stipendiyasi*. Class komponentalarini inkapsulyastiyalash, konstruktorlar va class metodlarini ishlab chiqing. Obyektlar massivini yarating va ular ustida qidirish, saralash, o'zgartirish amallarini bajaring.

Masalani yechish g'oyasi: *talaba fazosida* Talaba classi yaratiladi va unda maydonlar yopiq holda e'lon qilinadi. Konstruktor orqali talabaning familiyasi, ismi va kursi kiritiladi. Baholari massiv ko'rinishda e'lon qilinib, tasodifiy baholar oladigan metod yoziladi. Stipendiyasini baholariga qarab belgilovchi metod ham yaratiladi. Talaba sessiyani yopganligiga qarab kursdan kursga o'tkaziladi. Agar u 4 kurs bo'lsa bitiruvchi ekanligi haqida xabar chiqariladi. Buning uchun Talaba classida setKurs() metodi yoziladi. main funksiyasida familiya va ism uchun string massivi tuzilib (tashqaridan kiritmaslik uchun), Talaba classiga tegishli obyektlar massivining konstruktoriga parametr sifatida beriladi va talabalar haqida ma'lumot chop etiladi. main funksiyasida obyektlar massivi ustida talaba ismi bo'yicha qidiruvchi, familiyasi

bo'yicha tartiblovchi va kursini o'zgartiruvchi funksiyalar do'st funksiya qilib yoziladi. Natijada funksiya ichida to'g'ridan to'g'ri murojaat etiladi.

Dastur matni:

```
1. #include<iostream>
2. #include<string.h>
3. #include<stdlib.h>
4. #include<time.h>
5. using namespace std;
6. namespace talaba { // talaba nomli nonlar fazosi
7. class Talaba {
8. private: // modifikator
9. // yopiq maydonlar
10. string familiyasi, ismi;
11. int kursi, baholari[7];
12. int stipendiyasi;
13. public:
14. // sinfning yopiq maydonlariga kirish uchun do'st funksiyalar
15. friend void Qidirish(talaba::Talaba t[], int n);
16. friend void Saralash(talaba::Talaba t[], int n);
17. friend void Kurs_ozgartirish(talaba::Talaba t[], int n);
18. void set(string F, string I, int K) {
19. familiyasi = F;
20. ismi = I;
21. kursi = K;}
22. void get() {
23. cout<<"Fam: "<<familiyasi<<endl;
24. cout<<"Ismi: "<<ismi<<endl;
25. cout<<"Yoshi: "<<kursi<<endl;}
26. void setBaholari(int baho_soni) {
27. srand(time(NULL));
28. for (int i = 0; i <baho_soni; i++) {
29. baholari[i]=rand()%6+1;
30. }}
31. void setKursi() {
32. kursi++;}
33. void setStipendiyasi(int baho_soni) {
34. int minBaho=baholari[0];
35. for (int i = 0; i <baho_soni; i++) {
36. if(minBaho>baholari[i]) minBaho=baholari[i];}

37. switch (minBaho){
```

```

38.case 3: stipendiyasi=270000; break;
39.case 4: stipendiyasi=400000; break;
40.case 5: stipendiyasi=550000; break;
41.default:stipendiyasi=0; } } };
42.void Qidirish(talaba::Talaba t[], int n) {
43.    cout<<"Qidirilayotgan talaba ismini kiriting: "<<endl;
44.string s; cin>>s; bool bor=false;
45.for (int i = 0; i < n; i++) {
46.if(t[i].ismi.compare(s)==0) {bor=true;
47.t[i].get();} }
48.if(!bor) cout<<"Bu ismli talaba ro'yhatda yo'q!"<<endl;
49.cout<<endl; }
50.void Saralash(talaba::Talaba t[], int n) {
51.cout<<"Familiya bo'yicha saralangan holati: \n";
52.talaba::Talaba temp;
53.for (int i = 0; i <n ; i++) {
54.for (int j = 0; j <n ; j++) {
55.if(t[i].famiyasi.compare(t[j].famiyasi)<0) {
56.temp=t[i]; t[i]=t[j]; t[j]=temp; } } }
57.for (int i = 0; i <n ; i++) {
58.t[i].get(); } cout<<endl;}
59.void Kurs_ozgartirish(talaba::Talaba t[], int n) {
60.cout<<"Kursdan kursga o'tganligi haqida ma'lumot:\n";
61.bool bitiruvchi=false;
62.for (int i = 0; i <n ; i++) {
63.if(t[i].stipendiyasi!=0 && t[i].kursi==4)
64.bitiruvchi=true;
65.if(t[i].stipendiyasi!=0 && t[i].kursi<4)
66.t[i].setKursi();
67.if(!bitiruvchi) t[i].get();
68.else{
69.t[i].get();
70.cout<<" Siz bitiruvchisiz!!!"<<endl;}
71.bitiruvchi = false; } } }
72.int main(){
73.string fam[]={ "Xoshimov", "Saidaliyeva", "Baxodirov","Nishonov"};
74.string ism[]={ "Doston", "Xurshida", "Temur", "Botir"};
75.int yoshlari[]={18,19,20,21};
76.talaba::Talaba t[100];
77.for (int i = 0; i <4; i++) {
78.t[i].set(fam[i],ism[i],yoshlari[i]);
79.t[i].setBaholari(4);
80.t[i].setStipendiyasi(4);

```

```

81. t[i].get(); }
82. cout<<endl;
83. talaba::Qidirish(t,4);
84. talaba::Saralash(t,4);
85. talaba::Kurs_ozgartirish(t,4);
86.}

```

Dastur natijasi:

Fam: Xoshimov
Ismi: Doston
Yoshi: 18
Fam: Saidaliyeva
Ismi: Xurshida
Yoshi: 19
Fam: Baxodirov
Ismi: Temur
Yoshi: 20
Fam: Nishonov
Ismi: Botir
Yoshi: 21

Qidirilayotgan talaba ismini kiriting:

Temur
Fam: Baxodirov
Ismi: Temur
Yoshi: 20

Familiya bo'yicha saralangan holati:

Fam: Baxodirov
Ismi: Temur
Yoshi: 20

Fam: Nishonov
Ismi: Botir
Yoshi: 21
Fam: Saidaliyeva
Ismi: Xurshida
Yoshi: 19
Fam: Xoshimov
Ismi: Doston
Yoshi: 18

Kursdan kursga o'tganligi haqida ma'lumot:

Fam: Baxodirov
Ismi: Temur
Yoshi: 20
Fam: Nishonov
Ismi: Botir
Yoshi: 21
Fam: Saidaliyeva
Ismi: Xurshida
Yoshi: 19
Fam: Xoshimov
Ismi: Doston
Yoshi: 18

Laboratoriya mashg'ulotini bajarish bo'yicha variantlar

1. **Klavyatura** va **sichqoncha** class larini yarating va ularni bog'lovchi metodlar yarating
2. **Kompyuter monitori** va **video kartasini** class larini yarating va ularni bog'lovchi funksiyalar yarating
3. **Telefon** va **notebook** class larini yarating va ularni bog'lovchi funksiyalar yarating

4. **Universitet** va **kollej** class larini yarating va ularni bog'lovchi funksiyalar yarating
5. **Kollej** va **maktab** class larini yarating va ularni bog'lovchi funksiyalar yarating
6. **Piyola** va **choynak** class larini yarating va ularni bog'lovchi funksiyalar yarating
7. **Stol** va **stul** class larini yarating va ularni bog'lovchi funksiyalar yarating
8. **Ruchka** va **qalam** class larini yarating va ularni bog'lovchi funksiyalar yarating
9. **Kastyum** va **shim** class larini yarating va ularni bog'lovchi funksiyalar yarating
10. **Kitob** va **daftar** class larini yarating va ularni bog'lovchi funksiyalar yarating
11. **Eshik** va **deraza** class larini yarating va ularni bog'lovchi funksiyalar yarating
12. **Chelak** va **tog'ora** class larini yarating va ularni bog'lovchi funksiyalar yarating
13. **Kalit** va **qulf** class larini yarating va ularni bog'lovchi funksiyalar yarating
14. **Svet** va **sham** class larini yarating va ularni bog'lovchi funksiyalar yarating
15. **Dom** va **hovli** class larini yarating va ularni bog'lovchi funksiyalar yarating
16. **Muzlatgich** va **mikrovalovka** class larini yarating va ularni bog'lovchi funksiyalar yarating
17. **Printer** va **skanner** class larini yarating va ularni bog'lovchi funksiyalar yarating
18. **Mashina** va **motochikl** class larini yarating va ularni bog'lovchi funksiyalar yarating
19. **Ovqat** va **Ichimlik** class larini yarating va ularni bog'lovchi funksiyalar yarating
20. **Fleshka** va **disk** class larini yarating va ularni bog'lovchi funksiyalar yarating

21. **Tufli** va **tapichka** class larini yarating va ularni bog'lovchi funksiyalar yarating

22. **Inson** va **hayvon** class larini yarating va ularni bog'lovchi funksiyalar yarating

Takrorlash uchun savol va topshiriqlar:

1. Obyektlarga ixtisoslashgan dasturlashning afzalliklari nimalardan iborat?
2. Class deganda nima tushuniladi?
3. Class qanday e'lon qilinadi?
4. Metod nima vazifani bajaradi?
5. Metod qayerda e'lon qilinadi?
6. Lokal o'zgaruvchi deganda nima tushuniladi?
7. Namespace nima?
8. Namespace std nima?
9. Namespace tarkibi nimalardan tashkil topgan?
10. Nomlar fazosi yataish nima uchun kerak?
11. Namespace da classlar yaratish mumkinmi?
12. Namespace da funksiyalar yaratish mumkinmi?
13. Namespace da polimorfizmdan foydalanish mumkinmi?
14. Namespace da inkapsulyatsiyadan foydalanish mumkinmi?
15. Namespace konstruktorlar yaratish mumkinmi?
16. Sinflar bir biri bilan qanday bog'lanadi?
17. Sinflardan voris olishning maqsadi nima?
18. Do'st funksiyalar nimaga kerak?
19. Do'st sinflarning maqsadi nima?
20. Namespace da operatorlarni qayta yozish mumkinmi?

2- LABORATORIYA ISHI

2. STL kutubxonalari. Konteyner sinflar

2.1. Chiziqli konteynerlar (array, vector, deque, list, forward_list)

2.1.1. array sinfidan foydalanib, amaliy dasturlar yaratish

Ishning maqsadi: C++ dasturlash tilida **array** sinflaridan foydalanib, konteynerlar dan foydalanish ko'nikmalarini shakllantirish.

Masalaning qo'yilishi:

array asosida yaratilgan to'plamni kamayish va ko'payish tartibida saralash.

Masala: *Talaba* nomli class yaratish. Konteynerlarning array sinfi yordamida to'plamlar yaratish. **Array** ning maxsus metodidan foydalanib, talabalar to'plamini talabalarning yoshlari bo'yicha saralovchi dastur tuzish.

Masalani yechish g'oyasi: **talaba** nomli yangi nolar fazosi yaratiladi. Uning tarkibiga **Talaba** classi va **Ism, Fam, Yosh, Step** nomli to'plamlar joylashtiriladi. Talaba classining ruxsat berilmagan maydonlaridan foydalanish uchun friend funksiya yaratiladi.

Dastur matni:

```
1. #include "stdafx.h"
2. #include <string>
3. #include <iterator>
4. #include <iostream>
5. #include <algorithm>
6. #include <array>
7. using namespace std;
8. const int n=3;
9. array<string, n> Ism;
10. array<string, n> Fam;
11. array<int, n> Yosh;
12. array<float, n> Step;
13. namespace talaba_2{
14. class Talaba{
15. int yosh;
16. std::string ism;
```

```

17. std::string fam;
18. float step;
19. public: //Saralash uchun do'st funksiyadan foydalanish
20.     friend void Saralash_show(array<Talaba, n> &T);
21. void set(int Y, std::string I, std::string F, float S){
22.     static int i=0;
23.     Yosh[i] = Y; Ism[i] = I; Fam[i] = F; Step[i] = S;
24.     yosh = Y;
25.     ism = I;
26.     fam = F;
27.     step = S; i++;
28. }
29. void get(){
30.     cout<<"Ismi: "<<ism<<endl;
31.     cout<<"Familiyasi: "<<fam<<endl;
32.     cout<<"Yoshi: "<<yosh<<endl;
33.     cout<<"Stependiyasi: "<<step<<endl;
34. }; //
35. void show(array<Talaba, n> &T){
36.     for (int i = 0; i < n; i++)
37.     { cout<<i+1<<" - talaba haqida haqida"<<endl;
38.     T[i].get();}
39. } // Konteynerga qiymat o'zlashtirish
40. void SetArray(array<Talaba, n> &T){
41.     string I,F; int Y; float S;
42.     for (int i = 0; i < n; i++)
43.     {
44.     cout<<i+1<<" - talaba haqida kiriting"<<endl;
45.     cout<<"Ism: "; cin>>I; cin.ignore();
46.     cout<<"Fam: "; cin>>F;
47.     cout<<"Yosh: "; cin>>Y;
48.     cout<<"Step: "; cin>>S;
49.     T[i].set(Y,I,F,S);}
50. } //saralb chiqarish uchun funksiya
51. void Saralash_show(array<Talaba, n> &T){
52.     for (int j = 0; j < n; j++){
53.     for (int i = 0; i < n; i++){
54.     if(Yosh[j]==T[i].yosh){
55.     cout<<j+1<<" - talaba haqida haqida"<<endl;
56.     T[i].get();}     }} }
57. }
58. int main(){

```

```

59. // n ta elementdan iborat T nomli konteyner
60. array<talaba_2::Talaba, n> T;
61. T[0].set(22,"Islom","Salimov", 400000);
62. T[1].set(19,"Oqil","Qodirov", 600000);
63. T[2].set(20,"Komil","Tashev", 1200000);
64. talaba_2::show(T);
65. cout<<endl<<"Yoshlari bo'yicha saralanga talabalar
    ro'yxati"<<endl;
66. std::sort(Yosh.begin(), Yosh.end());
67. talaba_2::Saralash_show(T);
68. getchar();
69. }

```

Dastur natijasi:

1 - talaba haqida haqida
 Ismi: Islom
 Familiyasi: Salimov
 Yoshi: 22
 Stependiyasi: 400000
 2 - talaba haqida haqida
 Ismi: Oqil
 Familiyasi: Qodirov
 Yoshi: 19
 Stependiyasi: 600000
 3 - talaba haqida haqida
 Ismi: Komil
 Familiyasi: Tashev
 Yoshi: 20
 Stependiyasi: 120000

Yoshlari bo'yicha saralanga talabalar
 ro'yxati
 1 - talaba haqida haqida
 Ismi: Oqil
 Familiyasi: Qodirov
 Yoshi: 19
 Stependiyasi: 600000
 2 - talaba haqida haqida
 Ismi: Komil
 Familiyasi: Tashev
 Yoshi: 20
 Stependiyasi: 120000
 3 - talaba haqida haqida
 Ismi: Islom
 Familiyasi: Salimov
 Yoshi: 22
 Stependiyasi: 400000

Quyida keltirilgan 3- jadvaldagi vazifalarda kiruvchi ma'lumotlar **matinli fayldan** o'zlashtirilishi kerak. Natijalar ham **matinli faylga** saqlanishi kerak.

Laboratoriya mashg'ulotini bajarish bo'yicha variantlar

1- jadval

1	Talabalar haqida ma'lumotlar berilgan (class Talaba). Ixtiyoriy talabani (Fam, Iismi, Sharfi) bo'yicha qidirib topuvchi dastur tuzing.
2	Kitoblar haqida ma'lumotlar berilgan (class Kitob). Ixtiyoriy kitobni nomi, avtor va yili bo'yicha qidirib topuvchi dastur tuzing.

3	Stol va stullar haqida ma'lumotlar berilgan (class Sto_stul). Ixtiyoriy stol va stulni 6 ta parametri bo'yicha qidirib topuvchi dastur tuzing.
4	Sinf xonalari haqida ma'lumotlar berilgan (class Classroom). Ixtiyoriy sinfonani nomeri va band yoki band emasligi bo'yicha qidirib topuvchi dastur tuzing.
5	Avtomobillar haqida ma'lumotlar berilgan (class Avto). Ixtiyoriy avtomobilni 6 ta parametri bo'yicha saralovchi dastur tuzing.
6	Muzeylar haqida ma'lumotlar berilgan (class Museum). Eng ko'p eksponantlarga ega bo'lgan muzeyni qidirib topuvchi dastur tuzing.
7	Sport komplekslari haqida ma'lumotlar berilgan (class Sprt_complex). Tovarlari arzon sport kompleksini qidirib topuvchi dastur tuzing.
8	Mashxur insonlar haqida ma'lumotlar berilgan (class Famous_person). Yaratilgan kashfiyotlarning nomi bo'yicha Mashxur insonlarni qidirib topuvchi dastur tuzing.
9	Universitetlar haqida ma'lumotlar berilgan (class University). Ilmiy salohiyati eng yuqori 3 ta universitetni qidirib topuvchi dastur tuzing.
10	Litseylar haqida ma'lumotlar berilgan (class Lyceums). Olinpiadada 3 yil ichida g'olib bo'lgan Litseyini qidirib topuvchi dastur tuzing.
11	Mashxur futbolistlar haqida ma'lumotlar berilgan (class Famous_footballer). Eng ko'p urilgan gollar, sariq va qizil kartochkalar, chiroyli va muhim gollari bo'yicha mashxur futbalkchini qidirib topuvchi dastur tuzing.
12	Kompyuterlar haqida ma'lumotlar berilgan (class Computers). Ushbu kompyuterlarning qaiysi birlari kompyuter o'yinlarini qo'llab quvvatlashishini va ularning sonini aniqlovchi dastur tuzing.
13	Telefonlar haqida ma'lumotlar berilgan (class Telephone). Yadrolari va operativ xotirasi kuchli telefonni qidirib topuvchi dastur tuzing.
14	Restoranlar haqida ma'lumotlar berilgan (class Restaurants). Milliy taomlari ko'p restoranni qidirib topuvchi dastur tuzing.
15	Taomlar haqida ma'lumotlar berilgan (class Meal). Suyuq va qo'yi taomlarni nomalri asosida qidirib topuvchi dastur tuzing.
16	Hayvonlar haqida ma'lumotlar berilgan (class Hayvon). Yozziyi hayvonlarni turi bo'yicha topuvchi dastur tuzing.
17	Parrandalar haqida ma'lumotlar berilgan (class Parranda). Eng ko'p tuxum qo'yadigan parannalarni turi qidirib topuvchi dastur tuzing.
18	A'lochi talabalar haqida ma'lumotlar berilgan (class Alochilar). Fanni va undan olgan bahosi bo'yicha a'lochi talabani qidirib topuvchi dastur tuzing.
19	Uy hayvonlari haqida ma'lumotlar berilgan (class Uy_hayvon). Eng foydali uy hayvonini foydali jihatlari bo'yicha qidirib topuvchi dastur tuzing.
20	Kompyuter o'yin dasturlari haqida ma'lumotlar berilgan (class Games). O'ngni rivojlantiruvchi kompyuter o'yinini har xil parametrlar asosida qidirib topuvchi dastur tuzing.

21	Past o'zlashtiruvchi talabalar haqida ma'lumotlar berilgan (class Passiv). Eng past o'zlashtiruvchi talabani har xil parametrlari bo'yicha qidirib topuvchi dastur tuzing.
22	Viloyatlar haqida ma'lumotlar berilgan (class Viloyatlar). Ixtiyoriy viloyatni har xil parametrlar bo'yicha qidirib topuvchi dastur tuzing.

2.1.2. vector sinfidan foydalanib, amaliy dasturlar yaratish

Ishning maqsadi: C++ dasturlash tilida **vecot** sinflaridan foydalanib, konteynerlardan foydalanish ko'nikmalarini shakllantirish.

Masalaning qo'yilishi:

vector asosida yaratilgan to'plam elementlarini vectorning maxsus funksiyalari yordamida qayta ishlash.

Masala: Berilgan int turidagi to'plamda – ishorali elementlarning eng kattasini o'chiruvchi dastur tuzing.

Masalani yechish g'oyasi: talaba nomli yangi nomlar fazosi yaratiladi. Uning tarkibiga **Talaba** classi va **Ism, Fam, Yosh, Step** nomli to'plamlar joylashtiriladi. Talaba classining ruxsat berilmagan maydonlaridan foydalanish uchun friend funksiya yaratiladi.

Dastur matni:

```

1. #include "stdafx.h"
2. #include <iostream>
3. #include <vector>
4. using namespace System;
5. using namespace std;
6. namespace Vectorlar{ // nomlar sohasi yaratildi
7. int max,index;
8. //vektorga qiymat o'zlashtirish
9. void set_Vector(std::vector<int> &A, int n){
10.     for (int i = 0; i < n; i++)
11.         { int m; cout<<i+1<<" son: "; cin>>m;
12.         A.push_back(m);
13.         if(A[i]<0){ max = A[i];}}
```

```

14.     }//Vektor qiylatlarni chiqarish
15.     void get_Vector(std::vector<int> &A, int n){
16.     cout<<"To'plam elementlari"<<endl;
17.     for ( int k : A ) { std::cout << k << " ";
18.     } cout<<endl;
19.     } // vaektorning max elementini toppish funksiyasi
20.     void Max_Vector(std::vector<int> &A, int n){
21.     for (int i = 0; i < n; i++){
22.     if(A[i]<0 && max<A[i]){max = A[i]; index=i;} }
23.     cout<<"Max= " <<max<<endl;
24.     } // vektorning max elementini o'chirish
25.     void Delete_Max_Vector(std::vector<int> &A){
26.     A.erase(A.begin()+index);} }
27.     int main() {
28.     cout<<"Berilgan int turidagi to'plamda -ishorali
        elementlarning maksimalini o'chiruvchi dastur"<<endl;
29.     int n; cin>>n;
30.     std::vector<int> A;
31.     Vectorlar::set_Vector(A,n);
32.     Vectorlar::get_Vector(A,n);
33.     Vectorlar::Max_Vector(A,n);
34.     Vectorlar::Delete_Max_Vector(A);
35.     Vectorlar::get_Vector(A,n);
36.     getchar();
37.     }

```

Dastur natijasi:

Berilgan int turidagi to'plamdan ishorali elementlarning maksimalini o'chiruvchi dastur

4

1 son: 2

2 son: -4

3 son: -55

4 son: 3

To'plam elementlari

2 -4 -55 3

Max= -4

To'plam elementlari

2 -55 3

Laboratoriya mashg'ulotini bajarish bo'yicha variantlar

Quyidagi masalalardagi funksiyalar vector sinfidan foydalanilgan holda yechilishi kerak. Har-bir masalada kiritish va chiqarish funksiyalari hamda vector elementlarni fayldan yuklovchi va faylga saqlovchi funksiyalar bo'lishi shart.

1. Vector elementlarini max va min elementlarini o'chiruvchi funksiyalar yarating.
2. Berilgan N ta ixtiyoriy sonni max va min lari o'rtasidagi sonlarini o'chiruvchi funksiyalar yarating.
3. Vector ning 1- manfiy elementi bilan ohirgi manfiy elementi o'rtasidagi elementlarini chiqaruvchi va o'chiruvchi funksiyalar yarating.
4. Vector elementlari ichidan ixtiyoriy qidirilga son bor yo'qligini aniqlash va ushu sonni o'chiruvchi funksiyalari yarating.
5. Vectorni ixtiyoriy berilgan index gacha bo'lgan elementlarini o'chiruvchi funksiyalar yarating.
6. Vectorni ixtiyoriy qidirilgan elementigacha bo'lgan elementlarini o'chiruvchi funksiyalar yarating.
7. Vectorni eng katta elementigacha bo'lgan elementlarini o'chiruvchi funksiyalar yarating.
8. Vectorni eng kichik elementigacha bo'lgan elementlarini o'chiruvchi funksiyalar yarating.
9. Masiivni eng kichik elementidan keyingi elementlarini chiqaruvchi va o'chiruvchi funksiyalar yarating.
10. Vectorni eng katta elementidan keyingi elementlarini chiqaruvchi va o'chiruvchi funksiyalar yarating.
11. Vectorda qidirilgan ixtiyoriy son nechta ekanligini aniqlovchi va ushbu sonni o'chiruvchi funksiyalar yarating.
12. Vectorda bir xil elementlarini o'chiruvchi funksiyalar yarating.

13. Vectorda bir xil elementlar nechta ekanligini aniqlovchi va ushbu elementlarni o'chiruvchi funksiyalar yarating.
14. Ikkita ixtiyoriy vectorni bir xil ekanligini aniqlovchi va bir xil bo'lsa bittasini o'chiruvchi funksiyalar yarating.
15. Ikkita ixtiyoriy vectorlarni nechta elementi bir xil emasligini aniqlovchi va bir bo'lmaganlarini o'chiruvchi funksiyalar yarating.
16. A vectordan B vectorni tashkil qilish mumkin yoki mumkin emasligini aniqlovchi va tashkil qilib bo'lsa B vectorni o'chiruvchi funksiyalar yarating.
17. A vectordan B vectorni elementlarini o'chiruvchi funksiyalar yarating.
18. Vectorda eng ko'p takrorlangan elementlarini chiqaruvchi va o'chiruvchi funksiyalar yarating.
19. Vectorda eng kam takrorlangan elementlarini chiqaruvchi va o'chiruvchi funksiyalar yarating.
20. Vectordan ixtiyor index dagi elementini topuvchi va o'chiruvchi funksiyalar yarating.

2.1.3. deque konteyneridan foydalanib, amaliy dasturlar yaratish

Ishning maqsadi: C++ dasturlash tilida **deque** sinflaridan foydalanib, konteynerlardan foydalanish ko'nikmalarini shakllantirish.

Masalaning qo'yilishi:

deque asosida yaratilgan to'plam elementlarini **dequening** maxsus funksiyalari yordamida qayta ishlash.

Masala: Berilgan **string** turidagi to'plamda eng uzun so'zni belgilar sonini aniqlovchi dastur tuzing.

Masalani yechish g'oyasi: **Deque** nomli yangi nolar fazosi yaratiladi. Uning tarkibiga **Ism, Fam, Yosh, Step** nomli to'plamlar joylashtiriladi. Talaba classining ruxsat berilmagan maydonlaridan foydalanish uchun friend funksiya yaratiladi.

Dastur matni:

```
1. #include "stdafx.h"
2. #include <iostream>
3. #include <string>
4. #include <deque>
5. using namespace System;
6. using namespace std;
7. namespace Dequelar{
8. int max,index;
9. void set_Deque(std::deque<string> &D, int n){
10.     std::string s;
11.     for (int i = 1; i <=n; i++)
12.     { cout<<i<<" - satrni kiriting: "<<endl;
13.     getline(cin,s); s+=" ";
14.     D.push_back(s);}
15.     }
16. void get_Deque(std::deque<string> &D, int n){
17.     cout<<"To'plam elementlari"<<endl;
18.     for (int i = 0; i < n; i++)
19.     { cout<<i+1<<" - satr: "<<endl;
20.     cout<<"\t"<<D[i]<<endl; } cout<<endl;
21.     }
22. void Max_Deque(std::deque<string> &D, int n){
23.     int a[100]={0}, l=0;
24.     for (int i = 0; i < n; i++)
25.     { int k=0;
26.     for (int j = 0; j < D[i].size(); j++)
27.     {
28.     if(D[i][j].Equals(' ')) {
29.     a[l] = k; l++; k=-1;}
30.     cout<<D[i][j]; k++;}
31.     }
32.     cout<<endl;
33.     int max = a[0];
34.     cout<<"\n So'zlarning uzunliklari:"<<endl;
35.     for (int i = 0; i < l; i++){
36.     if(max<a[i]){max=a[i];}
37.     cout<<a[i]<<" ";}
38.     cout<<"Max= "<<max<<endl;}
39.     }
40. int main(){
```

```

41.     std::deque<std::string> D;
42.     cout<<"Berilgan string turidagi to'plamda eng uzun
        so'zni belgilar sonini aniqlovchi dastur"<<endl;
43.     int n; cout<<"To'plam elementlari soni n= ";
        cin>>n; cin.ignore();
44.     Dequelar::set_Deque(D,n);
45.     Dequelar::get_Deque(D,n);
46.     Dequelar::Max_Deque(D,n);
47.     getchar();
48.     }

```

Dastur natijasi:

Berilgan **string** turidagi to'plamda eng uzun so'zni belgilar sonini aniqlovchi dastur

To'plam elementlari soni n= 2

1 - satrni kiriting:

Mallayev Oybek Usmonqulovich

2 - satrni kiriting:

Mamatkarimov Sunnatullo Gayratjon o'g'li

To'plam elementlari

1 - satr:

Mallayev Oybek Usmonqulovich

2 - satr:

Mamatkarimov Sunnatullo Gayratjon o'g'li

Mallayev Oybek Usmonqulovich Mamatkarimov Sunnatullo Gayratjon o'g'li

So'zlarning uzunliklari:

8 5 13 12 10 9 6 Max= 13

Laboratoriya mashg'ulotini bajarish bo'yicha variantlar

Quyidagi masalalardagi funksiyalar **deque** sinfidan foydalanilgan holda yechilishi kerak. Har-bir masalada kiritish va chiqarish funksiyalari hamda deque elementlarini fayldan yuklovchi va faylga saqlovchi funksiyalar bo'lishi shart.

1. **String** turidagi **To'plam** berilgan. Undagi unli harflarni **char** turidagi **A** massivga, undoshlarini **B** massivga o'zlashtiruvchi dastur tuzing.

2. **String** turidagi **To‘plam** berilgan. Undagi ketma-ket kelgan unli harflarni **char** turidagi **A** massivga, undoshlarini **B** massivga o‘zlashtiruvchi dastur tuzing.
3. **String** turidagi **To‘plam** berilgan. Undagi elementlarni **char** turidagi **A** massivga joylashtirib, o‘shish tartibida saralovchi, **B** massivga kamayuvchi tartibida saralovchi dastur tuzing.
4. **String** turidagi **To‘plam** berilgan. Undagi elementlarni **char** turidagi **A** massivga joylashtirib, ixtiyoriy elementdan keyingi elementlarni chiqaruvchi dastur tuzing.
5. **String** turidagi **To‘plam** berilgan. Undagi elementlarni **char** turidagi **A** massivga joylashtirib, yangi metodning 2 ta parametrlari o‘rtasidagi elementlarni chiqaruvchi dastur tuzing. Agar bunday element mavjud bo‘lmasa “Bunday element mavjud emas” yozuvi chiqsin.
6. **String** turidagi **To‘plam** berilgan. Undagi elementlarni **char** turidagi **A** massivga joylashtirib, toq elementlarini **B** massivga, juftlarni **C** massivga joylashtiruvchi dastur tuzing.
7. **String** turidagi **To‘plam** berilgan. Undagi elementlarni **char** turidagi **A** massivga joylashtirib, tub elementlarini **B** massivga joylashtiruvchi dastur tuzing.
8. **String** turidagi **To‘plam** berilgan. Undagi elementlarni **char** turidagi **A** massivga joylashtirib, eng katta tub elementni topuvchi dastur tuzing.
9. **String** turidagi **To‘plam** berilgan. Undagi elementlarni **char** turidagi **A** massivga joylashtirib, so‘zlar, gaplar sonini aniqlovchi dastur tuzing.
10. **String** turidagi **To‘plam** berilgan. Undagi elementlarni **char** turidagi **A** massivga alfavit bo‘yicha, **byte** turidagi **B** massivga o‘shish tartibida joylashtiruvchi dastur tuzing.
11. **String** turidagi **To‘plam** berilgan. Undagi elementlarni **char** turidagi **A** massivga joylashtirib, ixtiyoriy to‘plam mavjudligini aniqlovchi dastur tuzing.

12. **String** turidagi **To‘plam** berilgan. Undagi elementlarni **char** turidagi **A** massivga joylashtirib, toq uzunlikdagi va juft uzunlikdagi so‘zlar sonini aniqlovchi dastur tuzing.
13. **String** turidagi **To‘plam** berilgan. Undagi elementlarni **char** turidagi **A** massivga joylashtirib, tub uzunlikdagi so‘zlar sonini aniqlovchi dastur tuzing.
14. **String** turidagi **To‘plam** berilgan. Undagi elementlarni **char** turidagi **A** massivga joylashtirib, polindrom so‘zlar sonini aniqlovchi dastur tuzing.
15. **String** turidagi **To‘plam** berilgan. Undagi elementlarni **char** turidagi **A** massivga joylashtirib, do‘st sonlar mavjudligini aniqlovchi dastur tuzing.
16. **String** turidagi **To‘plam** berilgan. Undagi elementlarni **char** turidagi **A** massivga joylashtirib, eng ko‘p takrorlangan harfni va sonini aniqlovchi dastur tuzing.
17. **String** turidagi **To‘plam** berilgan. Undagi elementlarni **char** turidagi **A** massivga joylashtirib, eng kam takrorlangan harfni va uni sonini aniqlovchi dastur tuzing.
18. **String** turidagi **To‘plam** berilgan. Undagi elementlarni **char** turidagi **A** massivga joylashtirib, eng ko‘p takrorlangan harflarni olib tashlovchi dastur tuzing.
19. **String** turidagi **To‘plam** berilgan. Undagi elementlarni **char** turidagi **A** massivga joylashtirib, **H – O** harflari oralig‘idagi harflarni olib tashovchi dastur tuzing.
20. **String** turidagi **To‘plam** berilgan. Undagi elementlarni **char** turidagi **A** massivga joylashtirib, so‘zlar uzunligi yig‘indisini aniqlovchi dastur tuzing.
21. **String** turidagi **To‘plam** berilgan. Undagi elementlarni **char** turidagi **A** massivga joylashtirib, bir xil uzunlikdagi so‘zlar sonini aniqlovchi dastur tuzing.
22. **String** turidagi **To‘plam** berilgan. Undagi elementlarni **char** turidagi **A** massivga joylashtirib, har - xil uzunlikdagi so‘zlar sonini aniqlovchi dastur tuzing.

2.1.4. forward_list konteyneridan foydalanib, amaliy dasturlar yaratish

Ishning maqsadi: C++ dasturlash tilida **forward_list** konteyneri va uning metodlaridan foydalanish ko'nikmalarini egallash.

Masalaning qo'yilishi:

Worward_list asosida yaratilgan to'plam elementlarini **forward_list** ning maxsus funksiyalari yordamida qayta ishlash.

Masala: Berilgan **int** turidagi to'plamdagi qiymatlarning raqamlari yig'indisini Z to'plamga joylashtiruvchi va ularni ekranga chiqaruvchi dastur tuzing.

Masalani yechish g'oyasi: **forward_list** konteyneri to'plami yaratiladi. **forward_list** ning **push_front()** iteratoridan foydalanib, A to'plamga qiymatlar o'zlashtiriladi. Z to'plamga esa A to'plam qiymatlarini raqamlari yig'indisi yoziladi.

Dastur matni:

```
1. #include "stdafx.h"
2. #include <iostream>
3. #include <forward_list>
4. #include "string";
5. #include <stdio.h>
6. using namespace System;
7. using namespace std;
8. int func(int d){
9. int s=0, k=0;
10. while(d>9){
11. s += d%10;
12. d=d/10; k++;}
13. return s+d;}
14. void Set_value(forward_list<int> &A, int n){
15. for (int i = 0; i < n; i++)
16. { float m; cout<<i+1<<" son: "; cin>>m;
17. A.push_front(m);}}
18. int main(){
19. cout<<"Forward_list"<<endl;
20. int n; cin>>n; cin.ignore();
21. std::forward_list<int> A;
22. std::forward_list<int> Z;
23. Set_value(A,n);
```

```

24.   auto s = A.begin();
25.   int d, k=0;
26.   cout<<"A to'plam qiymatlari"<<endl;
27.   while(s != A.end()){
28.     cout<<*s<<endl;
29.     d=*s;
30.     Z.push_front(func(d));
31.     s++; }
32.   cout<<"Z to'plam qiymatlari"<<endl;
33.   for (auto i=Z.begin(); i!=Z.end(); i++){
34.     cout<<*i<<endl;}
35.   getchar();
36.   }

```

Dastur natijasi:

Forward_list

2

1 son: 23

2 son: 45

A to'plam qiymatlari

45

23

Z to'plam qiymatlari

5

9

Laboratoriya mashg'ulotini bajarish bo'yicha variantlar

Quyida berilgan vazifalarda kiruvchi qiymatlar matnli fayldan o'zlashtirilishi kerak.

2- jadval

1	<i>U</i> to'plam berilgan. $[a,b]$, intervaliga tegishli bo'lgan elementlar sonini aniqlang va ularni <i>B</i> to'plamga qo'shadigan dastur tuzing.
2	<i>R</i> to'plam berilgan. <i>D</i> to'plamga uning maksimal va minimal qiymatlaridan tashqari bo'lgan qiymatlarini yozadigan dastur tuzing.
3	<i>X</i> to'plamining o'rta arifmetik va o'rta geometrik qiymatlarini aniqlang. <i>X</i> to'plamining o'rta arifmetik qiymatidan katta va o'rta geometrik qiymatdan kichik bo'lgan qiymatlar sonini aniqlovchi va chop etuvchi dastur tuzing.
4	<i>U</i> to'plamning musbat qiymatlarni, <i>B</i> to'plamning toq qiymatlari bilan almashtiruvchi dastur tuzing.

5	Z to‘planning manfiy juft qiytlarini B to‘planning musbat toq qiymatlaridan keyin joylashtiradigan dastur tuzing.
6	H to‘planning o‘rta arifmetik qiymatidan kichik bo‘lgan qiymatlarini B to‘plamga joylashtirib, H to‘plamdan o‘chiradigan dastur tuzing.
7	H to‘planning eng katta va eng kichik qiymatlarini toping va ularning o‘rtasidagi qiymatlarni o‘chiradigan dastur tuzing.
8	H to‘planning 1- va ohirgi manfiy qiymatlarini toping va ularning o‘rtasidagi qiymatlarni o‘chiradigan dastur tuzing.
9	X to‘plam berilgan. X to‘plam qiymatlarini kamayib borish tartibida joylang. Agar bir xil qiymatlilari bo‘lsa, ulardan bittasini qoldirib, qolganlarini o‘chiradigan dastur tuzing.
10	X to‘planning musbat qiymatlaridan Y to‘plamni tashkil eting, so‘ngra Y to‘plam qiymatlarini kamayib borish tartibida chiqaruvchi dastur tuzing.
11	X to‘planning tub qiymatlaridan D to‘plamni tashkil eting, so‘ngra X to‘plamdan ushbu tub qiymatlarini o‘chiruvchi dastur tuzing.
12	X to‘planning bir xil qiymatlilarini o‘chiruvchi dastur tuzing.
13	X va Y ikkita to‘plam berilgan. X va Y to‘plamlarining musbat qiymatlilaridan Z to‘plamni tashkil etuvchi dastur tuzing.
14	X to‘plam berilgan. Uning qiymatlari oldiga 3 ni joylashtirib(masalan: 4 bo‘lsa, 34. 5 bo‘lsa 35), ularni max va min larini aniqlovchi dastur tuzing.
15	X to‘plam berilgan. Uning juft indeksli qiymatlarini Z to‘plamga ko‘chirib yozing. Z to‘plamdagi maksimal va minimal elementlarining o‘rnini o‘zaro almashtirib, chiqaruvchi dastur tuzing.
16	X to‘plam berilgan. X to‘plamdagi manfiy qiymatlarni Z to‘plamga ko‘chirib yozing. Hosil bo‘lgan Z to‘planning max va min qiymatlaridan boshqa qiymatlarini ekranga chiqaruvchi dastur tuzing.
17	X to‘plam berilgan. X to‘plamdagi qiymatlarni tublarini Z to‘plamga yozing. Hosil bo‘lgan Z to‘planning max qiymatlaridan kichik va min qiymatlaridan 2 barobar kattalari ekranga chiqaruvchi dastur tuzing.
18	X to‘plam berilgan. X to‘plamdagi 6 xonali qiymatlarni raqamlari yig‘indisini Z to‘plamga yozing. Hosil bo‘lgan Z to‘planning max qiymatlaridan kichik va min qiymatlaridan 2 barobar kattalari ekranga chiqaruvchi dastur tuzing.
19	X to‘plam berilgan. X to‘plamdagi butun tipli qiymatlarni Z to‘plamga, haqiqiy tipli qiymatlari Y to‘plamga yozuvchi va ularni ekranga chiqaruvchi dastur tuzing.
20	X to‘plam berilgan. X to‘plamdagi 6 xonali qiymatlarni raqamlari yig‘indisi toq bo‘lsa Z to‘plamga yozuvchi va ularni ekranga chiqaruvchi dastur tuzing.
21	X to‘plam berilgan. X to‘plamdagi qiymatlarni nechta xonali ekanlini Z to‘plamga yozuvchi va ularni ekranga chiqaruvchi dastur tuzing.

22	<i>X</i> to‘plam berilgan. <i>X</i> to‘plamdagi qiymatlarni Ixtiyoriy beringan <i>M</i> sonidan kattalarini <i>Z</i> to‘plamga yozuvchi va ularni ekranga chiqaruvchi dastur tuzing.
23	<i>X</i> to‘plam berilgan. <i>X</i> to‘plamdagi qiymatlarni Ixtiyoriy beringan <i>M</i> sonidan kattalarini <i>Z</i> to‘plamga, kichiklarini <i>D</i> to‘plamga yozuvchi va ularni ekranga chiqaruvchi dastur tuzing.

2.1.5. list konteyneridan foydalanib, amaliy dasturlar yaratish

Ishning maqsadi: C++ dasturlash tilida **list** konteyneri va uning metodlaridan foydalanish ko‘nikmalarini egallash.

Masalaning qo‘yilishi:

list asosida yaratilgan to‘plam elementlarini **list** ning maxsus funksiyalari yordamida qayta ishlash.

Masala: Berilgan **int** turidagi to‘plam qiymatlarini juftlarini *Z* to‘plamga joylashtiruvchi va ularni ekranga chiqaruvchi dastur tuzing.

Masalani yechish g‘oyasi: **list** konteyneri to‘plami yaratiladi. **list** ning **push_back()** iteratoridan foydalanib, *A* to‘plamga qiymatlar o‘zlashtiriladi. *Z* to‘plamga esa *A* to‘plam qiymatlarini juftlari yoziladi.

Dastur matni:

```

1. #include "stdafx.h"
2. #include <iostream>
3. #include <list>
4. #include "string";
5. #include <stdio.h>
6. using namespace System;
7. using namespace std;
8. // Listga qiymat o‘zlashtirish
9. void Set_value(list<int> &A, int n){
10.     for (int i = 0; i < n; i++)
11.         { float m; cout<<i+1<<" son: "; cin>>m;
12.         A.push_back(m);} // Listga qiymat o‘zlashtirish
13.     }
14.     int main(){

```



```

15.     cout<<"List konteyneri"<<endl;
16.     int n; cin>>n; cin.ignore();
17.     std::list<int> A; // A nomi List konteyner
18.     std::list<int> Z;// Z nomi List konteyner
19.     Set_value(A,n); // funksiya parametrida konteyner
20.     auto s = A.begin();
21.     int d, k=0;
22.     cout<<"A to'plam qiymatlari"<<endl;
23.     while(s != A.end()){
24.         cout<<*s<<endl;
25.         d=*s; if(d%2==0)
26.             Z.push_back(d);
27.         s++; }
28.     cout<<"Z to'plam qiymatlari"<<endl;
29.     for (auto i=Z.begin(); i!=Z.end(); i++){
30.         cout<<*i<<endl;}
31.     getchar();
32.     }

```

Dastur natijasi:

List konteyneri	4
5	71
1 son: 56	13
2 son: 4	26
3 son: 71	Z to'plam qiymatlari
4 son: 13	56
5 son: 26	4
A to'plam qiymatlari	26
56	

4- jadvalda berilgan vazifalarni list konteynerida yordamida amalga oshirish kerak. Kiruvchi qiymatlar matnli fayldan o'zlashtiriladi.

2.2. Assotsiativ konteynerlar (set, map, multiset, multimap)

2.2.1. set va multiset sinflaridan foydalanib, amaliy dasturlar yaratish

Ishning maqsadi: C++ dasturlash tilida **set** va **multiset** sinflari va uning metodlaridan foydalanish ko'nikmalarini egallash.

Masalaning qo'yilishi:

set va **multiset** asosida yaratilgan to'plam elementlarini **set** va **multiset** ning maxsus funksiyalari yordamida qayta ishlash.

Masala:

Berilgan **int** turidagi to'plam qiymatlari 2- to'plamda nechta marta qatnashganligini aniqlovchi va ularni ekranga chiqaruvchi dastur tuzing. Ikkala to'plamdan ham bir xil qiymatli elementlar o'chirilib, saralangan to'plam hosil qilinsin hamda uning qiymatlari ekranga chiqarilsin.

Masalani yechish g'oyasi: **set** konteyneri to'plami yaratiladi. **list** ning **insert()** iteratoridan foydalanib, A to'plamga qiymatlar o'zlashtiriladi. Z to'plamga esa A to'plam qiymatlarini juftlari yoziladi.

Dastur matni:

```
1. #include "stdafx.h"
2. #include <set>
3. #include<iostream>
4. #include<stdlib.h>
5. #include<time.h>
6. using namespace System;
7. using namespace std;
8. int main()
9. { srand(time(NULL));
10.     set<int> s; int n;
11.     cout<<"Elementlar sonini: "; cin>>n;
12.     multiset<int> M, M2; // 2 ta konteyner yaratish
13.     for (int i = 0; i < n; i++)
14.     { int j = rand()%n+n; M.insert(j);
15.       j = rand()%n+n; M2.insert(j); }
16.     auto k2 = M.begin(); // auto turida o'zgaruvhchi
17.     set<int> s2; // set konteynerini yaratish
```

```

18.   for (int i = 0; i < n; i++){
19.   s2.insert(*k2); k2++; // set ga qiymat kiritish}
20.   cout<<endl<<"M to'plam elementlari:"<<endl;
21.   for (auto i=M.begin(); i!=M.end(); i++){
22.   cout<<*i<<" ";} cout<<endl;
23.   cout<<endl<<"M2 to'plam elementlari:"<<endl;
24.   for (auto i=M2.begin(); i!=M2.end(); i++){
25.   cout<<*i<<" ";}
26.   cout<<endl<<endl<<endl<<"M1 ning elementlari M2
    to'plamda qatnashganlari soni:"<<endl;
27.   int soni=0;
28.   //s2 ning dastlabki elementidan ohirigacha
29.   for (auto i=s2.begin(); i!=s2.end(); i++){
30.   for (auto j=M2.begin(); j!=M2.end(); j++){
31.   if(M2.count(*i)){
32.   if(*i==*j) {soni++; }} }
33.   if(soni>0)cout<<*i<<" -"<<soni<<" marta "<<endl;
34.   else {cout<<*i<<" - qatnashmagan "<<endl;}
35.   soni = 0;}
36.   auto k = M.begin(); auto l = M2.begin();
37.   for (int i = 0; i < n; i++){
38.   s.insert(*k);k++; s.insert(*l); l++; } cout<<endl;
39.   cout<<endl<<"Saralangan to'plam elementlari:"<<endl;
40.   for (auto i=s.begin(); i!=s.end(); i++) {
41.   cout<<*i<<" "; }getchar();
42.   }

```

Dastur natijasi:

Elementlar sonini: 25	30 -1 marta
M to'plam elementlari:	31 -4 marta
25 27 27 27 27 28 29 30 31 31 31	32 - qatnashmagan
32 33 33 35 35 37 37 40 42 42 43	33 - qatnashmagan
43 47 48	35 -1 marta
M2 to'plam elementlari:	37 -1 marta
26 27 30 31 31 31 31 34 35 37 38	40 - qatnashmagan
39 42 43 45 45 45 46 46 46 46 47	42 -1 marta
47 49 49	43 -1 marta
M1 ning elementlari M2 to'plamda	47 -2 marta
qatnashganlari soni:	48 - qatnashmagan
25 - qatnashmagan	
27 -1 marta	Saralangan to'plam elementlari:
28 - qatnashmagan	25 26 27 28 29 30 31 32 33 34 35
29 - qatnashmagan	37 38 39 40 42 43 45 46 47 48 49

Laboratoriya mashg'ulotini bajarish bo'yicha variantlar

Quyida berilgan vazifalarda kiruvchi qiymatlar matnli fayldan o'zlashtirilishi kerak.

3- jadval

1	Berilgan int turidagi to'plam qiymatlarining juftlari 2- to'plamda nechta marta qatnashganligini aniqlovchi va ularni ekranga chiqaruvchi dastur tuzing. Ikkala to'plamdan ham bir xil qiymatli elementlar o'chirilib Saralangan to'plam hosil qilinsin hamda uning qiymatlari ekranga chiqarilsin.
2	Berilgan int turidagi to'plam qiymatlarining toqlari 2- va 3- to'plamda nechta marta qatnashganligini aniqlovchi va ularni ekranga chiqaruvchi dastur tuzing. Ikkala to'plamdan ham bir xil qiymatli elementlar o'chirilib Saralangan to'plam hosil qilinsin hamda uning qiymatlari ekranga chiqarilsin.
3	Berilgan int turidagi to'plam qiymatlarining tublari 2- to'plamda nechta marta qatnashganligini aniqlovchi va ularni ekranga chiqaruvchi dastur tuzing. Ikkala to'plamdan ham bir xil qiymatli elementlar o'chirilib Saralangan to'plam hosil qilinsin hamda uning qiymatlari ekranga chiqarilsin.
4	Berilgan float turidagi to'plam qiymatlarining butun(nuqtadan oldin) qismi 2- to'plamga, kasr(nuqtadan keyin) qismi 3- to'plamga joylashtiruvchi va ularni ekranga chiqaruvchi dastur tuzing. Ikkala to'plamdan ham bir xil qiymatli elementlar o'chirilib Saralangan to'plam hosil qilinsin hamda uning qiymatlari ekranga chiqarilsin.
5	Berilgan float turidagi to'plamning 2 ta elementi yig'indisidan 2- to'plamni, ayirmasidan 3- to'plamni hosil qiluvchi va ularni ekranga chiqaruvchi dastur tuzing. Ikkala to'plamdan ham bir xil qiymatli elementlar o'chirilib Saralangan to'plam hosil qilinsin hamda uning qiymatlari ekranga chiqarilsin.
6	Berilgan float turidagi to'plamning 2 ta juft elementi yig'indisidan 2- to'plamni, ayirmasidan 3- to'plamni hosil qiluvchi va ularni ekranga chiqaruvchi dastur tuzing. Ikkala to'plamdan ham bir xil qiymatli elementlar o'chirilib Saralangan to'plam hosil qilinsin hamda uning qiymatlari ekranga chiqarilsin.
7	Berilgan string turidagi to'plamning 2 ta elementidan 2- to'plamni, 3 ta elementidan 3- to'plamni hosil qiluvchi va ularni ekranga chiqaruvchi dastur tuzing. Ikkala to'plamdan ham bir xil qiymatli elementlar o'chirilib Saralangan to'plam hosil qilinsin hamda uning qiymatlari ekranga chiqarilsin.
8	Berilgan string turidagi to'plamning juft uzunlikka ega bo'lgan elementidan 2- to'plamni, toq uzunlikdagi elementidan 3- to'plamni hosil qiluvchi va ularni ekranga chiqaruvchi dastur tuzing. Ikkala to'plamdan ham bir xil qiymatli elementlar o'chirilib Saralangan to'plam hosil qilinsin hamda uning qiymatlari ekranga chiqarilsin.
9	Berilgan string turidagi to'plamning 2 ta so'zdan iborat elementidan 2- to'plamni, 1 ta so'zdan iborat elementidan 3- to'plamni hosil qiluvchi va ularni ekranga chiqaruvchi dastur tuzing. Ikkala to'plamdan ham bir xil

	qiymatli elementlar o'chirilib Saralangan to'plam hosil qilinsin hamda uning qiymatlari ekranga chiqarilsin.
10	Berilgan string turidagi to'plamning 1 ta so'zdan iborat elementlarini o'chiruvchi va 1 ta so'zdan iborat elementidan 2- to'plamni hosil qiluvchi hamda ularni ekranga chiqaruvchi dastur tuzing.
11	Berilgan string turidagi to'plamning kata harfdan iborat elementlarini o'chiruvchi va kichik harfdan iborat elementidan 2- to'plamni hosil qiluvchi hamda ularni ekranga chiqaruvchi dastur tuzing.
12	Berilgan string turidagi to'plamda kata harfdan va kichik harfdan boshlanuvchi elementlari soni aniqlovchi hamda ularni ekranga chiqaruvchi dastur tuzing.
13	Berilgan string turidagi to'plamning kata harfdan iborat elementlarini kichik harfga o'girib, 2- to'plamni va kichik harfdan iborat elementlarini katta harfga o'girib, 3- to'plamni hosil qiluvchi hamda ularni ekranga chiqaruvchi dastur tuzing.
14	Berilgan char turidagi to'plamning 2 ta elementidan 2- to'plamni, 3 ta elementidan 3- to'plamni hosil qiluvchi va ularni ekranga chiqaruvchi dastur tuzing.
15	Berilgan char turidagi to'plamning juft uzunlikka ega bo'lgan elementidan 2- to'plamni, toq uzunlikdagi elementidan 3- to'plamni hosil qiluvchi va ularni ekranga chiqaruvchi dastur tuzing.
16	Berilgan char turidagi to'plamning 2 ta so'zdan iborat elementidan 2- to'plamni, 1 ta so'zdan iborat elementidan 3- to'plamni hosil qiluvchi va ularni ekranga chiqaruvchi dastur tuzing.
17	Berilgan char turidagi to'plamning 1 ta so'zdan iborat elementlarini o'chiruvchi va 1 ta so'zdan iborat elementidan 2- to'plamni hosil qiluvchi hamda ularni ekranga chiqaruvchi dastur tuzing.
18	Berilgan char turidagi to'plamning kata harfdan iborat elementlarini o'chiruvchi va kichik harfdan iborat elementidan 2- to'plamni hosil qiluvchi hamda ularni ekranga chiqaruvchi dastur tuzing.
19	Berilgan char turidagi to'plamda kata harfdan va kichik harfdan boshlanuvchi elementlari soni aniqlovchi hamda ularni ekranga chiqaruvchi dastur tuzing.
20	Berilgan char turidagi to'plamning kata harfdan iborat elementlarini kichik harfga o'girib, 2- to'plamni va kichik harfdan iborat elementlarini katta harfga o'girib, 3- to'plamni hosil qiluvchi hamda ularni ekranga chiqaruvchi dastur tuzing.
21	Berilgan char turidagi to'plamning elementlarini ASCII kodlaridan 2- to'plamni va element so'zlarining teskasidan(masalan: Oybek - kebyO) o'girib, 3- to'plamni hosil qiluvchi hamda ularni ekranga chiqaruvchi dastur tuzing.

2.2.2. map va multimap sinflaridan foydalanib, amaliy dasturlar yaratish

Ishning maqsadi: C++ dasturlash tilida **map** va **multimap** sinflari va uning metodlaridan foydalanish ko'nikmalarini egallash.

Masalaning qo'yilishi:

map va **multimap** asosida yaratilgan to'plam elementlarini **map** va **multimap** ning maxsus funksiyalari yordamida qayta ishlash.

Masala:

Talabalar haqida (familiya, ismi va stipendiyasi) **string** va **float** turidagi to'plam berilgan. Familiyasi yoki ismi to'liq yozilganlar va stipendiyasi N ga teng bo'lganlaridan 2- to'plamni hosil qiluvchi va ularni ekranga chiqaruvchi dastur tuzing.

Masalani yechish g'oyasi:

map konteyneri to'plamni yaratiladi. **map** ning `insert(pair<char,int>('a',10))` iteratoridan foydalanib, A to'plamga qiymatlar o'zlashtiriladi.

Dastur matni:

```
1. #include "stdafx.h"
2. #include <map>
3. #include<iostream>
4. #include<string>
5. using namespace System;
6. using namespace std;
7. int main()
8. {   map<string, int> s; int n;
9.   cout<<"Talabalar sonini kiriting: "; cin>>n;
      cin.ignore();
10.   multimap<string,int> M, M2;
11.   for (int i = 0; i < n; i++)
12.   {
13.     string St;
14.     int step;
15.     cout<<i+1<<" - talaba ma'lumotlarini
      kiriting:"<<endl;
```



```

16.     cout<<"Fam, ismi: "; getline(cin,St);
17.     cout<<"Stipendiyasi: "; cin>>step; cin.ignore();
18.     s.insert(pair<string, int>(St,step));
19.     M.insert(pair<string, int>(St,step));
20.     }
21.     string S;
22.     int N;
23.     cout<<"Qanday stipendiya oladigan talaba haqida
        ma'lumot kerak?: ";
24.     cin>>N;
25.     int k=0;
26.     for (auto it = M.begin(); it != M.end(); ++it){
27.         S = it->first;
28.         if(S.find(" ")<S.length() && it->second == N){
29.             M2.insert(pair<string, int>(S,N));
                cout<<"bor"<<endl;}
30.     }
31.     cout<<"So'rov Natijasi:"<<endl;
32.     for (auto it = M2.begin(); it != M2.end(); ++it){
33.         cout<<it->first<<" "<<it->second<<endl;
34.     }
35.     getchar(); getchar();
36.     return 0;
37.     }

```

Dastur natijasi:

Talabalar sonini kiriting: 3

Fam, ismi: ABdurahmonov

1 - talaba ma'lumotlarini kiriting:

Stipendiyasi: 300000

Fam, ismi: Mallayev Oybek

Qanday stipendiya oladigan talaba

Stipendiyasi: 400000

haqida ma'lumot kerak?: 400000

2 - talaba ma'lumotlarini kiriting:

bor

Fam, ismi: Ishniyazov Odil

So'rov Natijasi:

Stipendiyasi: 500000

Mallayev Oybek 400000

3 - talaba ma'lumotlarini kiriting:

Laboratoriya mashg'ulotini bajarish bo'yicha variantlar

Quyida berilgan vazifalarda kiruvchi qiymatlar matnli fayldan o'zlashtirilishi kerak.

4- jadval

1	Talabalar haqida (familiya, ismi, sharfi va telefon) to'plam berilgan. FIO si to'liq yozilganlar va telefoni T ga teng bo'lganlaridan 2- to'plamni hosil qiluvchi va ularni ekranga chiqaruvchi dastur tuzing.
2	Talabalar haqida (familiya, ismi, sharfi, guruhi va bahosi) to'plam berilgan. Bahosi 5 bo'lganlaridan 2- to'plamni, bahosi 4 bo'lganlaridan 3- to'plamni, bahosi 3 bo'lganlaridan 4- to'plamni va fanni o'zlashtira olmaganlaridan 5- to'plamni hosil qiluvchi va 5 ta to'plamni ekranga chiqaruvchi dastur tuzing.
3	Talabalar haqida (familiya, ismi, sharfi, viloyat, tuman va qishloq) to'plam berilgan. Har bir viloyatda nechta talaba borligini aniqlovchi va viloya, tuman bo'yich talabalarni qidiruvchi dastur tuzing.
4	Talabalar haqida (familiya, ismi, sharfi, fani, guruhi va bahosi) to'plam berilgan. Ixtiyoriy qidirilayotgan guruhda har bir fandan nechta talaba 5 bahoga, nechtasi 4 bahoga, nechtasi 3 bahoga va nechtasi o'zlashtira olamayotganligi aniqlovchi va bahosi bo'yich talabalarni qidiruvchi dastur tuzing.
5	Kompyuterlar haqida (nomi, firmasi, xotirasi, yadrosi, yili va yadrosi) to'plam berilgan. Kompyuterlarni yadrosi va firmasi bo'yicha qidiruvchi hamda topilgan natijani faylga saqlashni taklif qiladigan dastur tuzing.
6	Kompyuter viruslari haqida (nomi, zarar darajasi, tarqalish tezligi, turi, va internet_id) to'plam berilgan. Kompyuter viruslarini nomi va tarqalish tezligi bo'yicha qidiruvchi hamda topilgan natijani faylga saqlashni taklif qiladigan dastur tuzing.
7	Kafedralar haqida (nomi, mudir_FIO, xodimlar soni, fanlar soni va professorlar soni) to'plam berilgan. Kafedralarni nomi, mudir_FIO, fanlari soni va professorlar soni bo'yicha qidiruvchi hamda topilgan natijani faylga saqlashni taklif qiladigan dastur tuzing.
8	Universitetlar haqida (nomi, rektor_FIO, xodimlar soni, fanlar soni, talabalar soni va professorlar soni) to'plam berilgan. Universitetlarni nomi, rektor_FIO, xodimlar soni, fanlar soni, talabalar soni va professorlar soni bo'yicha qidiruvchi hamda topilgan natijani faylga saqlashni taklif qiladigan dastur tuzing.
9	TTJ lar haqida (nomi, xokim_FIO, xonalar soni, talabalar soni, qizlar va bollar soni hamda nogironlar soni, nomeri) to'plam berilgan. TTJ ni nomi, xokim_FIO, xonalar soni, talabalar soni, qizlar va bollar soni hamda

	nogironlar soni, nomeri bo'yicha qidiruvchi hamda topilgan natijani faylga saqlashni taklif qiladigan dastur tuzing.
10	Maktablar haqida (nomi, direktor_FIO, xodimlar va o'qituvchilar soni, qizlar va bollar soni, olinpiada g'oliblari soni hamda maktab nomeri) to'plam berilgan. Maktablarni nomi, direktor_FIO, xodimlar va o'qituvchilar soni, qizlar va bollar soni, olinpiada g'oliblari soni hamda maktab nomeri bo'yicha qidiruvchi hamda topilgan natijani faylga saqlashni taklif qiladigan dastur tuzing.
11	Fanalar haqida (nomi, o'qituvchi_FIO, 5 chilar soni, 4 chilar soni, 3 chilar soni va o'zlashtira olmaganlar soni) to'plam berilgan. Fanlarni nomi, o'qituvchi_FIO, 5 chilar soni, 4 chilar soni, 3 chilar soni va o'zlashtira olmaganlar soni bo'yicha qidiruvchi hamda topilgan natijani faylga saqlashni taklif qiladigan dastur tuzing.
12	Operatsion tizimlar haqida (nomi, yili, razryadlar soni, turi va imkoniyatlarining soni) to'plam berilgan. Operatsion tizimlarni nomi, yili, razryadlar soni, turi va imkoniyatlarining soni bo'yicha qidiruvchi hamda topilgan natijani faylga saqlashni taklif qiladigan dastur tuzing.
13	Avtomobillar haqida (nomi, yili, versiyasi va narxi) to'plam berilgan. Avtomobillarni nomi, yili, versiyasi va narxi bo'yicha qidiruvchi hamda topilgan natijani faylga saqlashni taklif qiladigan dastur tuzing.
14	Ho'l mavalar haqida (nomi, mazasi, rangi, turi va narxi) to'plam berilgan. Ho'l mevalarni nomi, mazasi, rangi, turi va narxi bo'yicha qidiruvchi hamda topilgan natijani faylga saqlashni taklif qiladigan dastur tuzing.
15	Korxonalar xodimlari haqida (FIO, yili, staji, lavozimi va oyligi) to'plam berilgan. Xodimlarning familiyasi bir xillardan 2- to'plamni, ismi bir xillardan 3- to'plamni, staji bir xillardan 4- to'plamni va oyligi bir xillardan 5- to'plamni tashkil qiluvchi hamda xodimni FIO, yili, staji, lavozimi va oyligi bo'yicha qidiruvchi hamda topilgan natijani faylga saqlashni taklif qiladigan dastur tuzing.
16	Mashhur futbolchilar haqida (FIO, kuboklar soni, staji, mashxurlik darajasi va oyligi) to'plam berilgan. Futbolchilarning familiyasi bir xillardan 2- to'plamni, ismi bir xillardan 3- to'plamni, staji bir xillardan 4- to'plamni va oyligi bir xillardan 5- to'plamni tashkil qiluvchi hamda futbolchini FIO, kuboklar soni, staji, mashxurlik darajasi va oyligi bo'yicha qidiruvchi hamda topilgan natijani faylga saqlashni taklif qiladigan dastur tuzing.
17	Kompyuter o'yinlari haqida (nomi, turi, versiyasi, hajmi, zararli oqibatlarini, foydalilik darajasi va aktuallik darajasi) to'plam berilgan. Kompyuter o'yinlarining nomi bir xillardan 2- to'plamni, zararli oqibatlarini bir xillardan 3- to'plamni, aktuallik darajasi bir xillardan 4- to'plamni tashkil qiluvchi hamda Kompyuter o'yinini nomi, turi, versiyasi, hajmi,

	zararli oqibatlarini, foydalilik darajasi va aktuallik darajasi bo'yicha qidiruvchi hamda topilgan natijani faylga saqlashni taklif qiladigan dastur tuzing.
18	Mobil telefonlar haqida (nomi, turi, versiyasi, hajmi, foydalilik darajasi, yadrosi, kamera o'lchami) to'plam berilgan. Mobil telefonlarning nomi bir xillardan 2- to'plamni, turi bir xillaridan 3- to'plamni, yadrosi bir xillaridan 4- to'plamni tashkil qiluvchi hamda Mobil telefonni nomi, turi, versiyasi, hajmi, foydalilik darajasi, yadrosi, kamera o'lchami bo'yicha qidiruvchi hamda topilgan natijani faylga saqlashni taklif qiladigan dastur tuzing.
19	Sinf xonalar haqida (nomeri, qavati, Javobgar shaxs_FIO, stollar soni, stellar soni, dars bo'lish vaqtlari va kompyuterlar soni) to'plam berilgan. Sinf xonalarning nomeri va qavati bir xillaridan 2- to'plamni, Javobgar shaxs_FIO si bir xillaridan 3- to'plamni, stollar soni bir xillaridan 4- to'plamni, stellar soni bir xillaridan 5- to'plamni, kompyuter soni bir xillaridan 6- to'plamni tashkil qiluvchi hamda ixtiyoriy sana va vaqtda bo'sh bo'lgan xonalarni aniqlovchi dastur tuzing. Topilgan natijani faylga saqlashni tashkil qilsin.
20	O'qituvchilar haqida (FIO, qabul qilish vaqti, fan nomi, lavozimi va fanlar soni) to'plam berilgan. O'qituvchilarning familiyasi va ismi bir xillaridan 2- to'plamni, qabul qilish vaqti bir xillaridan 3- to'plamni, lavozimi bir xillaridan 4- to'plamni, fanlar soni bir xillaridan 5- to'plamni tashkil qiluvchi hamda ixtiyoriy sana va vaqtda nechta va qaysi o'qituvchilar talabalarni qabul qilishi mumkinligini aniqlovchi dastur tuzing. Topilgan natijani faylga saqlashni tashkil qilsin.
21	Kitoblar haqida (nomi, avtor_FIO, yili, yo'nalishi, sahifalar soni va foydalanilganlik soni) to'plam berilgan. Kitoblarning avtor_FIO si bir xillaridan 2- to'plamni, yili bir xillaridan 3- to'plamni, yo'nalishi bir xillaridan 4- to'plamni, sahifalar soni bir xillaridan 5- to'plamni tashkil qiluvchi hamda eng ko'p va eng kam o'qilgan kitoblarni va ularning sonini aniqlovchi dastur tuzing. Topilgan natijani faylga saqlashni tashkil qilsin.
22	Arxiv fayllari haqida (nomi, turi, javobgar_FIO, yili, o'zgartirilgan sanasi, saqlash muddati) to'plam berilgan. Arxiv fayllarning javobgar_FIO si bir xillaridan 2- to'plamni, yili bir xillaridan 3- to'plamni, o'zgartirilgan sanasi bir xillaridan 4- to'plamni, saqlash muddati bir xillaridan 5- to'plamni tashkil qiluvchi hamda saqlash muddati tugagan fayllarni va ularning sonini aniqlovchi dastur tuzing. Topilgan natijani faylga saqlashni tashkil qilsin.

2.3. stack, queue, priority_queue konteyner adapterlari

2.3.1. stack sinflaridan foydalanib, amaliy dasturlar yaratish

Ishning maqsadi: C++ dasturlash tilida **stack** sinflari va uning metodlaridan foydalanish ko'nikmalarini egallash.

Masalaning qo'yilishi:

stack asosida yaratilgan to'plam elementlarini **push()**, **top()**, **pop()** va **empty()** iteratorlar yordamida qayta ishlash.

Masala:

Savatga hajmi har xil (bir xillari ham bor) tuxumlar solingan. Ushbu tuxumlardan bir xil hajimli tuxumlar sonini aniqlovchi va ulardan bittasini qoldirish va qolganlarini o'chirish dasturini tuzing.

Masalani yechish g'oyasi:

stack konteyneri to'plami yaratiladi. **Stack** ning **push()** iteratoridan va **set** ning **insert(<int>(a))** iteratoridan foydalanib, **Stack** to'plamga qiymatlar o'zlashtiriladi. Ichma ich sikl jarayonidan bir xil hajimli tuxumlar soni aniqlanadi.

Dastur matni:

```
1. #include "stdafx.h"
2. #include <stack>
3. #include <set>
4. #include<iostream>
5. #include<string>
6. #include<stdlib.h>
7. #include<time.h>
8. using namespace System;
9. using namespace std;
10. int main()
11. { srand(time(NULL));
12. int n; cout<<"Tuxumlar sonini kiriting:= "; cin>>n;
13. stack<int> Stak[100]; // int tipida stak yaratish
14. stack<int> Stak2[100];
15. set<int> M; // int tipida set konteyneri
16. for (int j = 0; j < n; j++) {
17. int a; a=rand()%n+n;;
```

```

18.   Stak[j].push(a); Stak2[j].push(a);
19.   M.insert(a);} cout<<endl<<endl;
20.   int w=0, sum=0;
21.   while (!Stak2[w].empty()){ int r=0;
22.   while (!Stak2[r].empty()){
23.   if(Stak2[w].top() == Stak2[r].top() && w!=r){
24.   sum+=1; Stak2[r].pop();
25.   } else r++; } w++; } int k=0;
26.   cout<<" Joriy qiymatlar"<<endl;
27.   while (!Stak[k].empty()){
28.   cout<< Stak[k].top()<<" "; k++;}
29.   cout<< endl<<" Bir xil qiymatli bo'lganlarini olib
      tashlanganlari "<<endl;
30.   for (auto it = M.begin(); it!=M.end(); it++){
31.   cout<<*it<<" "; } cout<<endl<<endl;
32.   cout<<"bir xil hajimli tuxumlar soni ="<<sum<<endl;
33.   getchar();
34.   }

```

Dastur natijasi:

Tuxumlar sonini kiriting:= 16

Joriy qiymatlar

24 18 16 18 24 20 25 26 24 28 23 20 30 29 27 29

Bir xil qiymatli bo'lganlarini olib tashlanganlari

16 18 20 23 24 25 26 27 28 29 30

bir xil hajimli tuxumlar soni = 2

Laboratoriya mashg'ulotini bajarish bo'yicha variantlar

Quyida berilgan vazifalarda kiruvchi qiymatlar matnli fayldan o'zlashtirilishi kerak.

5- jadval

1	Savatga har xil rangli (bir xillari ham bor) tuxumlar solingan. Ushbu tuxumlardan bir xil rangli tuxumlar sonini aniqlovchi va ulardan bittasini qoldirish va qolganlarini o'chirish dasturini tuzing.
---	--

2	Go'sht olish navbatda N ta talaba turibdi. Lekin ayrim ayyor talabalar navbatga 2 va undan ko'proq o'z nomini yozdirib qo'ygan. Ushbu navbatga 1 tadan ko'p yozilgan talabalarni va ularning sonini aniqlovchi dastur tuzing.
3	Go'sht olish uchun navbatda N ta talaba turibdi. Lekin ayrimlarning familiyasi bir xil. Shundan foydalanib talabalar navbatga 2 va undan ko'proq o'z nomini yozdirib qo'ygan. Ushbu navbatga familiyasi bir xil bo'lganlarni, ularning sonini aniqlovchi va ushbu bir xil familiyalilarga go'shtni teng taqsimlovchi dastur tuzing.
4	Navbatda yuvilish uchun N ta tarelka turibdi. Ichida yuvilganlari ham bor. Har bir tarelkaga 1 minutdan vaqt ketmoqda. Lekin tarelka yuvuvchining vaqti kam. U tarelkalarni $N/2$ vaqt birligida yuvib tugatmoqchi. Tarelka yuvuvchiga yordam beradigan dastur tuzing.
5	To'yda N ta qo'shnilardan M tadan kosalar olib chiqilgan. To'ydan keyin hammasi bitta yashikda solingan va ular raqamlangan. Har bir qo'shnidan nechtdan ($M=?$) kosa chiqqanini va ularni alohida ekranga chiqaruvchi dastur tuzing.
6	To'yda N ta qo'shnilardan M tadan kosalar olib chiqilgan. To'ydan keyin hammasi bitta yashikda solingan va ular raqamlangan. Har bir qo'shnining nechtdan ($M-K?$) kosa singanligini aniqlovchi va ularni alohida ekranga chiqaruvchi dastur tuzing.
7	Haydovchi yo'lovchini o'z manziliga yetkazib olib borishi kerak. Lekin yo'l triband. Sitaforning Yashil rangi yongunicha 1 dan N sekund o'tmoqda. N gacha bo'lgan sekundda qizil mashinalar ketma – ket kelishsa sitaforning qizil rangi yonmasdan yana Yashil rang 1 dan N gacha yonib turaverar ekan. Yo'lning boshlanishiga shuning uchun ham bir xil rangli mashinalar ketma ket ket kelsin deb yozib qo'yilgan ekan. Mashinalar ranglarini o'zgartirish imkoniyati mavjudligidan foydalanib, yo'lovchini qisqa vaqt ichida manziliga yetkazuvchi dastur tuzing.
8	Yashikda N ta xil rangli M ta koptoklar bor. Koptoklar raqamlangan. N ta xil koptoklarni ranglari va raqamlari bo'yich K ta yashiklarga saralab taxlaydigan dastur tuzing.
9	9 etajli, 4 ta padezdan iborat binoning M- qavatidan pastida ($1 < M \leq 9$) 1 ta xonodondan suv toshmoqda. Agar suvsoz hamma xonodonga kirib chiqsa suv o'zidan pastki hamma etajga tarqalib ketadi. xonodonni qisqa vaqt ichida topuvchi dastur tuzing.
10	Kitob javonda kitoblar betartib turibdi. Har bir kitob foydalanilganda unga 1 raqam qo'shib boriladi. N ta kitob M marta o'qilgandan keyin eng ko'p va eng kam o'qilgan kitoblarni va umuman o'qilmagan kitoblarni saralab beruvchi dastur tuzing.
11	Yashikda qizil, ko'k va qora rangli N ta ruchka bor. Har bir ruhkadan K marta foydalanilgan. Qizil, ko'k va qora rangli ruchkalarni eng ko'p va eng kam foydalanilganlarini topuvchi dastur tuzing.

12	C++ fanidan mustaqil ishni topshirishning ohirgi kunida N ta talaba navbatda turibdi. O'qituvchi joriy talabadan oldingi ikkita talabaga bergan savolini qo'shib, kelgan talabaga berganmoqda. M ta talabadan keyin o'qituvchi nechta savol berganligini aniqlovchi dastur tuzing.
13	Osh yeyish uchun N ta talaba dasturxonga o'tirdi. Talabalarnin yoshi har xil (teng yoshlilari ham bor). Bir kg osh K qoshiq oshga teng. O'zbeklarda ovqatlanish avval kattalardan boshlanadi. Shuning uchun kata yoshdagi talabalar yeyishni boshlaydi. Har bir talaba D ta qoshikdan osh yeydi. Lekin R yoshdagi M- talabaga osh yetmay qolishi mumkin. M- talabaga osh yatish yoki yatmasligini aniqlovchi dastur tuzing.
14	Qizil, ko'k va yashil rangli tufli ketma-ket tarqatilmoqda. Uni olish uchun N ta talaba navbatda turibdi. Lekin navbati kelgan talaba tufli rangini tanlash imkinoyati yoq. Qaysi rangdagi tufli ketma-ketligi to'g'ri kelsa shuni olib ketyapti. M- talaba ko'k tufli olish uchun joyini nechta marta boshqa talaba bilan almashtirish kerakligini aniqlovchi dastur tuzing.
15	N ta talaba stipendiya olish uchun navbatda turibdi. Har bir talabaga K minut vaqt sarflanmoqda. Kassa M vaqtda yopilishi shart. D vaqtda L- talaba stipendiya olishiga vaqt yetish yoki yetmasligini aniqlovchi dastur tuzing.
16	Kassadan M ta bilet sotilgan. Lekin kino teatrda kino ko'rish uchun bilet bilan navbatda N ta odam turibdi ($N < M$). K- qatorda nechta joy borligini aniqlovchi dastur tuzing. Biletlar hammasi sotilgan yoki sotilmagan bo'lishi ham mumkin.
17	Futbolchilar o'zlari tanlagan raqamlari bilan ketma-ket kamayish tartibida safda turibdi. Lekin ularning yoshlari bir biridan farq qiladi (tenglari ham bor). M- futbolist yosh bo'yicha saftda tursa, safning nechinchi o'rnida turishini, yoki raqamlari bo'yicha safta tursa, safning nechinchi o'rnini band qilishini aniqlovchi dastur tuzing.
18	Imtixon uchun o'qituvchi N ta talabaga bilet tayyorladi. Lekin imtixon paytida bilet tarqatish uchun vaqt kamligi uchun biletlarni aralashtirib tarqatdi. Aniqrog'i jurnalda 1- raqamli talabaga 1- variant tushishi kerak edi. Shunda o'qituvchi jurnalda qaysi talaba imtixonga kelmaganligini aniqlab olardi. Qaysi talaba imtixonga kelmaganligini aniqlovchi dastur tuzing.
19	Kredit tizimida darsga talabalar har xil guruxlardan kelganligi uchun bir birlarini yaxshi tanishmaydi. Shuning uchun o'zining o'rniga boshqa fanni yaxshi biladigan or'tog'ini kirtib yuboradi. Imtixonlar kompyuterda o'tkaziladi. Savollarni ochish uchun FIO kiritilgandan keyin, talabaning barmog' izlari tekshiriladi. Agar mos kelsa savollar ochiladi. Aks holda ochilmaydi. Imtixonga o'rtog'ini uchun kirga begona talabalarni aniqlovchi dastur tuzing.
20	Bitta guruhdagi talabalar qilgan xizmatlari uchun guruh rahbari tomonidan taqdirlanmoqda. Avval eng ko'p xizmat qilganlarni keyin esa qolganlarni taqdirlashga yordamlashuvchi dastur tuzing.

2.3.2. queue va priority_queue sinflaridan foydalanib, amaliy dasturlar yaratish

Ishning maqsadi: C++ dasturlash tilida **queue** va **priority_queue** sinflari va uning metodlaridan foydalanish ko'nikmalarini egallash.

Masalaning qo'yilishi:

queue va **priority_queue** asosida yaratilgan to'plam elementlarini **push()**, **top()**, **front()**, **back()** va **empty()** iteratorlar yordamida qayta ishlash.

Masala:

Imtixon nizomida formasiz kelgan talabalarni imtixonga chetlatish bandi mavjud. Lekin ayrim talabalar ush qoidaga amal qilamydilar. O'qituvchi eshik oldidida bittadan talaba formasini tekshirib, xonaga kiritmoqda. Guruhda N ta talaba mavjud. Imtixondan keyin imtixonga umuman kelmagan talabalar "men imtixonga kelgan edim lekin siz kiritadingiz" degan vaj bilan o'qituvchi ustidan shikoyat qilmasligi uchun nechta talaba imtixonga keldi, nechtasi formasiz keldi va nechtasi umuman kelmaganligini aniqlovchi dastru tuzing.

Masalani yechish g'oyasi:

queue va **priority_queue** konteynerlari to'plami yaratiladi. **queue** va **priority_queue** ning **push()** iteratoridan foydalanib, **queue** va **priority_queue** to'plamga qiymatlar o'zlashtiriladi. Ichma ich sikl jarayonidan foydalanib, bir xil hajmli imtixonga formasiz kelgan talabalar soni aniqlanadi. Umumiy talabalar sonidan formasiz kelgan talabalar soni ayrilib imtixonga umuman kelmagan talabalar soni aniqlanadi.

Dastur matni:

1. `#include "stdafx.h"`
2. `#include <stack>`
3. `#include <queue>`
4. `#include<iostream>`
5. `#include<string>`
6. `#include<stdlib.h>`
7. `#include<time.h>`

```

8. using namespace System;
9. using namespace std;
10. class Talaba{
11. int yosh; std::string ism; std::string fam;
12. float step; bool imtixon;
13. public:
14. bool get_imtixon(){return imtixon;}
15. void set(int Y, std::string I, std::string F, float S,
    bool Im){
16. yosh = Y; ism = I; fam = F; step = S; imtixon = Im;}
17. void get(){
18. cout<<"Ismi: "<<ism<<endl;
19. cout<<"Familiyasi: "<<fam<<endl;
20. cout<<"Yoshi: "<<yosh<<endl;
21. cout<<"Stependiyasi: "<<step<<endl;
22. cout<<"Formaning mavjudligi: "<<imtixon<<endl; } };
23. int main(){
24. int n; cout<<"Talabalar sonini kiriting:= ";
25. cin>>n; cin.ignore();
26. queue<Talaba> Navbat[100];
27. Talaba t[100];
28. string I,F; int Y; float S; bool Im=false;
29. for (int i = 0; i < n; i++){
30. cout<<i+1<<" - talaba haqida kiriting"<<endl;
31. cout<<"Ism: "; cin>>I;
32. cout<<"Fam: "; cin>>F;cin.ignore();
33. cout<<"Yosh: "; cin>>Y;
34. cout<<"Step: "; cin>>S;cin.ignore();
35. cout<<"Formaning bor yo'qligi: 0 yo'q, bor 1: ";
36. cin>>Im; t[i].set(Y,I,F,S,Im);
37. Navbat[i].push(t[i]); }
38. cout<<endl<<endl<<"Imtixonga formasiz kelgan
    talabalar"<< endl;
39. int f_soni=0;
40. for (int i = 0; i < n; i++){
41. if(t[i].get_imtixon() == 0){
42. cout<<endl<<f_soni+1<<" - talaba haqidagi
    ma'lumotlar"<<endl; f_soni++;
43. Navbat[i].front().get();} }
44. cout<<"Formasiz kelgan talabalar soni:
    "<<f_soni<<endl<<endl;

```

```

45. cout<<endl<<endl<<"Imtixonga formada kelgan
    talabalar"<< endl; f_soni=0;
46. for (int i = 0; i < n; i++){
47. if(t[i].get_imtixon() == 1){
48. cout<<endl<<f_soni+1<<" - talaba haqidagi
    ma'lumotlar"<<endl;
49. f_soni++; Navbat[i].front().get();} }
50. cout<<"Formada kelgan talabalar soni:
    "<<f_soni<<endl<<endl;
51. cout<<endl<<endl<<"Imtixonga jami kelgan talabalar
    ro'yhati"<< endl;
52. for (int i = 0; i < n; i++){
53. cout<<endl<<i+1<<" - talaba haqidagi
    ma'lumotlar"<<endl;
54. Navbat[i].front().get();}
55. cout<<"Guruhdagi jami talabalar soni:
    "<<n<<endl<<endl; getchar(); }

```

Dastur natijasi:

Talabalar sonini kiriting:= 3

1 - talaba haqida kiriting

Ism: Oybek

Fam: Mallayev

Yosh: 28

Step: 500000

Formaning bor yo'qligi: 0 yo'q, bor

1: 1

2 - talaba haqida kiriting

Ism: Akbar

Fam: Sattarov

Yosh: 18

Step: 750000

Formaning bor yo'qligi: 0 yo'q, bor

1: 0

3 - talaba haqida kiriting

Ism: Odil

Fam: Ishniyazov

Yosh: 20

Step: 560000

Formaning bor yo'qligi: 0 yo'q, bor

1: 1

Imtixonga formasiz kelgan talabalar

1 - talaba haqidagi ma'lumotlar

Ismi: Akbar

Familiyasi: Sattarov

Yoshi: 18

Stependiyasi: 750000

Formaning mavjudligi: 0

Formasiz kelgan talabalar soni: 1

Imtixonga formada kelgan talabalar

1 - talaba haqidagi ma'lumotlar

Ismi: Oybek

Familiyasi: Mallayev

Yoshi: 28

Stependiyasi: 500000

Formaning mavjudligi: 1

2 - talaba haqidagi ma'lumotlar

Ismi: Odil

Familiyasi: Ishniyazov
Yoshi: 20
Stependiyasi: 560000
Formaning mavjudligi: 1
Formada kelgan talabalar soni: 2

Imtixonga jami kelgan talabalar ro'yhati

1 - talaba haqidagi ma'lumotlar
Ismi: Oybek
Familiyasi: Mallayev
Yoshi: 28
Stependiyasi: 500000
Formaning mavjudligi: 1

2 - talaba haqidagi ma'lumotlar
Ismi: Akbar
Familiyasi: Sattarov
Yoshi: 18
Stependiyasi: 750000
Formaning mavjudligi: 0

3 - talaba haqidagi ma'lumotlar
Ismi: Odil
Familiyasi: Ishniyazov
Yoshi: 20
Stependiyasi: 560000
Formaning mavjudligi: 1
Guruhdagi jami talabalar soni: 3

2.4. Iteratorlar va ularning qo'llanilishi. Xotirani taqsimlovchilar va ularga qo'yilgan talablar

2.4.1. Standart algoritmlar va iteratorlardan foydalanib, amaliy dasturlar yaratish

Ishning maqsadi: C++ dasturlash tilida **Standart algoritmlar va iteratorlar** sinflari va uning qayta yuklangan turlaridan foydalanish ko'nikmalarini egallash.

Masalaning qo'yilishi:

Standart algoritmlar va iteratorlar asosida yaratilgan to'plam elementlarini

copy(from, from_end, to),

copy_if(from, from_end, to, pred),

copy_n(from, count, to),

copy_backward(from, from_end, to_end),

move(from, from_end, to),

move_backward(from, from_end, to_end),

swap_ranges(a, a_end, b),

fill(to, to_end, value), fill_n(to, n, value),
generate(to, to_end, gen),
generate(v.begin(), v.end(), rand),
generate_n(to, n, gen), reverse(begin, end),
reverse_copy(from, from_end, to),
find(begin, end, value),
mismatch(a, a_end, b, b_end, eq),
max_element(begin, end, comp),
min_element(begin, end, comp),
for_each(begin, end, fun),
replace(begin, end, old_val,
new_val), remove(begin, end, value),
remove_if(begin, end, pred),
sort(begin, end, comp),
iteratorlar yordamida qayta ishlash.

Masala:

Matnli faylda N ta talabaning familiyasi, ismi va yoshi yozilgan. Ushbu ma'lumotlarni to'plamga o'zlashtirib yoshi bo'yicha saralovchi dastru tuzing.

Masalani yechish g'oyasi:

Talaba haqida har xil turdagi ma'lumotlarni boshqarish uchun "**Talaba**" nomli struktura yaratiladi. **istream_iterator** iteratoridan foydalanib, matnli fayldan ma'lumotlar o'zlashtiriladi va **ostream_iterator** iteratoridan foydalanib, ma'lumotlarni matnli faylga yoki natijalar oynasiga chiqarish mumkin. Yosh bo'yicha saralash uchun **sort()**, ma'lumotlarni **vector** ga joylashtirish uchun **copy()**, ixtiyoriy yoshdan ixtiyoriy yoshgacha ma'lumotlarni chiqarish uchun **copy_if()** algoritmlaridan foydalaniladi. "**Talaba**" sun'iy tur bo'lganligi uchun kiritish va chiqarish oqimlari(>>, <<) qayta yuklanadi.

Dastur matni:

```

1. #include "stdafx.h"
2. #include <algorithm>
3. #include <vector>
4. #include <set>
5. #include <queue>
6. #include<iostream>
7. #include<string>
8. #include<stdlib.h>
9. #include<time.h>
10. #include<fstream>
11. using namespace System;
12. using namespace std;
13. struct Talaba{ // Talaba nomli struktura
14. std::string firstname, secondname;
15. size_t age;
16. };
17. // "<" amallarni qayta yuklash
18. bool comparator(const Talaba& p1, const Talaba& p2){
19. return p1.age < p2.age; }
20. std::ostream& operator << (std::ostream& out, const
    Talaba& p){
21. out << p.firstname << " " << p.secondname << " " <<
    p.age;
22. return out;
23. }
24. // ">>" amallarni qayta yuklash
25. std::istream& operator>>(std::istream& in, Talaba& p){
26. in >> p.firstname >> p.secondname >> p.age;
27. return in; }
28. // standart algoritmlarda parameter uchun qo'llashga
29. struct predicate{
30. size_t begin, end;
31. predicate(int p1, int p2): begin(p1), end(p2) {}
32. bool operator()(const Talaba &p){
33. return (p.age > begin) && (p.age < end); }
34. };
35. // Matnli faylga bo'g'lanish
36. void File_out_in(){
37. std::ifstream fin("input.txt");
38. std::ofstream fout("output.txt");
39. if(fin==NULL){cout<<"fayl not found"; return;}
40. std::vector<Talaba> v; // Talaba tipida v konteyner

```

```

41. // copy algoritmidan foydalanib, fayldan ma'lumot
    o'zlashtirish
42. std::copy(std::istream_iterator<Talaba>(fin),
    std::istream_iterator<Talaba>(), std::inserter(v,
    v.end()));
43. // sarta yoqdamida saralash
44. std::sort(v.begin(), v.end(), comparator);
45. // copy algoritmidan foydalanib, faylga ma'lumotlarni
    yozish
46. std::copy_if(v.begin(), v.end(),
    std::ostream_iterator<Talaba>(std::cout, "\n"),
    predicate(20, 25));
47. std::copy(v.begin(), v.end(),
    std::ostream_iterator<Talaba>(fout, "\n"));
48. //std::copy(v.begin(), v.end(),
    std::ostream_iterator<Talaba>(cout, "\n"));
49. }
50. int main(){
51. File_out_in(); getchar(); }

```

Dastur natijasi:

Natijalar oynasi

Salimov Nuriddin 21
 Sharipov Komil 21
 Mallayev Oybek 22
 Holiqov Abdulla 23
 Abdullayev Jalil 24

Input.txt fayli fayl

Mallayev Oybek 22
 Salimov Nuriddin 21
 Holiqov Abdulla 23
 Abdullayev Jalil 24
 Sharipov Komil 21

Yuqorida keltirilgan algoritmlarning baza birlariga misollar keltiramiz:

2.4.2. reverse_copy algoritmiga misol:

```

1. #include "stdafx.h"
2. // reverse_copy ga misollar
3. #include <iostream> // std::cout
4. #include <algorithm> // std::reverse_copy
5. #include <vector> // std::vector
6. using namespace std;
7. int main () {
8. int Int_massiv[] ={1,2,3,4,5,6,7,8,9};
9. vector<int> myvector;
10. myvector.resize(9); // joy ajratish

```

```

11. reverse_copy (Int_massiv, Int_massiv+9,
    myvector.begin());
12. // tarkibni chop qilish:
13. cout << "myvector tarkibi:";
14. for (vector<int>::iterator it=myvector.begin();
    it!=myvector.end(); ++it)
15. cout << ' ' << *it;
16. cout << '\n';
17. getchar();
18. return 0;
19. }

```

Dastur natijasi: myvector tarkibi: 9 8 7 6 5 4 3 2 1

2.4.3. find(begin, end, value) algoritmiga misol:

```

1. #include "stdafx.h"
2. // find misol
3. #include <iostream> // std::cout
4. #include <algorithm> // std::find
5. #include <vector> // std::vector
6. using namespace std;
7. int main () {
8. int Int_massiv[] = { 10, 20, 30, 40 };
9. int * p;
10. p = find (Int_massiv, Int_massiv+4, 30);
11. if (p != Int_massiv+4)
12. cout<<"Int_massiv da element topilgan: "<<*p<<'\n';
13. else
14. cout<<"Int_massiv da element topilmadi \n";
15. vector<int> myvector (Int_massiv,Int_massiv+4);
16. vector<int>::iterator it;
17. it = find (myvector.begin(), myvector.end(), 30);
18. if (it != myvector.end())
19. cout<<"myvector da element topildi: "<< *it <<'\n';
20. else
21. cout << "myvector da element topilmadi \n";
22. getchar(); return 0;
23. }

```

Dastur natijasi:

Int_massiv da element topilgan: 30

myvector da element topildi: 30

2.4.4. mismatch(a, a_end, b, b_end, eq) algoritmiga misol:

```
1. #include "stdafx.h"
2. #include <iostream>      // std::cout
3. #include <algorithm>    // std::mismatch
4. #include <vector>      // std::vector
5. #include <utility>     // std::pair
6. bool mypredicate (int i, int j) {
7. return (i==j); }
8. using namespace std;
9. int main () {
10. vector<int> myvector;
11. for (int i=1; i<6; i++) myvector.push_back (i*10);
12. // myvector: 10 20 30 40 50
13. int Int_massiv[] = {10,20,80,320,1024};
14. // Int_massiv: 10 20 80 320 1024
15. pair<vector<int>::iterator,int*> mypair;
16. // standart taqqoslash yordamida:
17. mypair = mismatch (myvector.begin(), myvector.end(),
18. Int_massiv);
19. cout << "Birinchi mos kelmaydigan elementlar: " <<
20. *mypair.first;
21. cout << " va " << *mypair.second << '\n';
22. ++mypair.first; ++mypair.second;
23. // predikt taqqoslash yordamida:
24. mypair = std::mismatch (mypair.first, myvector.end(),
25. mypair.second, mypredicate);
26. cout << "Ikkinchi mos kelmaydigan elementlar: " <<
27. *mypair.first;
28. cout << " va " << *mypair.second << '\n';
29. getchar(); return 0;
30. }
```

Dastur natijasi:

Birinchi mos kelmaydigan elementlar: 30 va 80

Ikkinchi mos kelmaydigan elementlar: 40 va 320

2.4.5. `max_element(begin, end, comp)` va `min_element(begin, end, comp)` algoritmiga misol:

```
1. #include "stdafx.h"
2. #include <iostream> // std::cout
3. #include <algorithm> // std::min_element,
   std::max_element
4. bool myfn(int i, int j) { return i<j; }
5. struct myclass {
6. bool operator() (int i,int j) { return i<j; }
7. } myobj;
8. using namespace std;
9. int main () {
10. int Int_massiv[] = {3,7,2,5,6,4,9};
11. // standart taqqoslash yordamida:
12. cout << "Eng kichik element - " <<
   *min_element(Int_massiv,Int_massiv+7) << '\n';
13. cout << "Eng katta element - " <<
   *max_element(Int_massiv,Int_massiv+7) << '\n';
14. // myfn funktsiyasidan komp sifatida foydalanish:
15. cout << "Eng kichik element " <<
   *min_element(Int_massiv,Int_massiv+7,myfn) << '\n';
16. cout << "Eng katta element " <<
   *max_element(Int_massiv,Int_massiv+7,myfn) << '\n';
17. // obyekt sifatida myobj-dan komp sifatida
   foydalanish:
18. cout << "Eng kichik element " <<
   *min_element(Int_massiv,Int_massiv+7,myobj) << '\n';
19. cout << "Eng katta element " <<
   *max_element(Int_massiv,Int_massiv+7,myobj) << '\n';
20. getchar(); return 0;
21. }
```

Dastur natijasi:

Eng kichik element - 2

Eng katta element - 9

Eng kichik element 2

Eng katta element 9

Eng kichik element 2

Eng katta element 9

2.4.6. For_each(begin, end, fun) algoritmiga misol:

```
1. #include "stdafx.h"
2. #include <iostream>      // std::cout
3. #include <algorithm>    // std::for_each
4. #include <vector>       // std::vector
5. void myfunction (int i) { // function:
6. std::cout << ' ' << i;
7. }
8. struct myclass {        // function object type:
9. void operator() (int i) {std::cout << ' ' << i;}
10. } myobject;
11. using namespace std;
12. int main () {
13. vector<int> myvector;
14. myvector.push_back(10);
15. myvector.push_back(20);
16. myvector.push_back(30);
17. cout << "myvector tarkibi:";
18. for_each (myvector.begin(), myvector.end(),
19.           myfunction);
19. cout << '\n';
20. cout << "myvector contains:";
21. for_each (myvector.begin(), myvector.end(), myobject);
22. cout << '\n';
23. getchar(); return 0;
24. }
```

Dastur natijasi:

myvector tarkibi: 10 20 30

myvector contains: 10 20 30

2.4.7. replace(begin, end, old_val, new_val) algoritmiga misol:

```
1. #include "stdafx.h"
2. #include <iostream>      // std::cout
3. #include <algorithm>    // std::replace
4. #include <vector>       // std::vector
5. using namespace std;
6. int main () {
7. int Int_massiv[] = { 10, 20, 30, 30, 20, 10, 10, 20 };
8. vector<int> myvector (Int_massiv, Int_massiv+8);
```

```

9. // 10 20 30 30 20 10 10 20
10. replace (myvector.begin(), myvector.end(), 20, 99);
11. // 10 99 30 30 99 10 10 99
12. cout << "myvector tarkibi:";
13. for (vector<int>::iterator it=myvector.begin();
    it!=myvector.end(); ++it)
14. cout << ' ' << *it;
15. cout << '\n';
16. getchar();
17. return 0;
18.}

```

Dastur natijasi: myvector tarkibi: 10 99 30 30 99 10 10 99

2.4.8. remove va remove_if(begin, end, pred) algorimiga misol:

```

1. #include "stdafx.h"
2. #include <iostream> // std::cout
3. #include <algorithm> // std::remove va remove_if
4. bool IsOdd (int i) { return ((i%2)==1); }
5. using namespace std;
6. int main () {
7. int Int_massiv[]={1,2,3,4,5,6,7,8,9};
8. int* pbegin = Int_massiv; // ^
9. int* pend = Int_massiv+sizeof(Int_massiv)/sizeof(int);
10. pend = remove_if (pbegin, pend, IsOdd);
11. cout << " remove_if dan keyin Range trakibi:";
12. for (int* p=pbegin; p!=pend; ++p)
13. cout << ' ' << *p; cout << '\n';
14. int Int_massiv_2[] = {3,2,3,4,3,6,7,3,9};
15. pbegin = Int_massiv_2;
16. int* pend_2 =
    Int_massiv_2+sizeof(Int_massiv_2)/sizeof(int);
17. pend_2 = remove (pbegin, pend_2, 3);
18. cout << " remove dan keyin Range trakibi:";
19. for (int* p=pbegin; p!=pend_2; ++p)
20. cout << ' ' << *p; cout << '\n';}

```

Dastur natijasi:

remove_if dan keyin Range trakibi: 2 4 6 8

remove dan keyin Range trakibi: 2 4 6 7 9

2.4.9. sort(begin, end, comp) algoritmiga misol:

```
1. #include <iostream>      // std::cout
2. #include <algorithm>     // std::sort
3. #include <vector>        // std::vector
4. bool myfunction (int i,int j) { return (i<j); }
5. struct myclass {
6. bool operator() (int i,int j) { return (i<j);}}
7. myobject;
8. using namespace std;
9. int main () {
10. int Int_massiv[] = {32,71,12,45,26,80,53,33};
11. vector<int> myvector (Int_massiv, Int_massiv+8);
12. // 32 71 12 45 26 80 53 33
    // standart taqqoslash (operator <) yordamida:
13. sort (myvector.begin(), myvector.begin()+4);
14. //(12 32 45 71)26 80 53 33
15. // funksiyadan foydalanib
16. sort (myvector.begin()+4, myvector.end(), myfunction);
17. // 12 32 45 71(26 33 53 80)
18. // objectdan foydalib
19. sort (myvector.begin(), myvector.end(), myobject);
20. //(12 26 32 33 45 53 71 80)
21. // elementlarni chop qilish:
22. cout << "myvector tarkitibi:";
23. for (vector<int>::iterator it=myvector.begin();
    it!=myvector.end(); ++it)
24. cout << ' ' << *it; cout << '\n'; getchar();}
```

Dastur natijasi:

myvector tarkitibi: 12 26 32 33 45 53 71 80

Laboratoriya mashg'ulotini bajarish bo'yicha variantlar.

Ushbu jadvaldagi vazifalar standart algoritmlardan foydalanilgan holda bajariladi.

6- jadval

No	Masalani berilishi
1.	Matnli faylda N ta talabaning familiyasi, ismi va yoshi yozilgan. Ushbu ma'lumotlarni to'plamga o'zlashtirib, ismi bo'yicha saralovchi dastur tuzing.

2.	Matnli faylda N ta stolning nomi, narxi va hajmi yozilgan. Ushbu ma'lumotlarni to'plamga o'zlashtirib, narxi bo'yicha saralovchi dastur tuzing.
3.	Matnli faylda N ta talabaning familiyasi, ismi va bahosi yozilgan. Ushbu ma'lumotlarni to'plamga o'zlashtirib, 5, 4 va 3 bahoga o'qiydigan talabalar soni aniqlaydigan dastur tuzing.
4.	Matnli faylda N ta talabaning familiyasi, ismi va yoshi yozilgan. Ushbu ma'lumotlarni to'plamga o'zlashtirib, yoshi eng katta va eng kichkina talabani aniqlaydigan dastur tuzing.
5.	Matnli faylda N ta talabaning familiyasi, ismi va yoshi yozilgan. Ushbu ma'lumotlarni to'plamga o'zlashtirib, Familiyasi A dan boshlanadigan talabalarni aniqlaydigan dastur tuzing.
6.	Matnli faylda N ta talabaning familiyasi, ismi va yoshi yozilgan. Ushbu ma'lumotlarni to'plamga o'zlashtirib, Ismi B dan boshlanadigan talabalarni va Familiyasi C dan boshlanadiganlarni aniqlaydigan dastur tuzing.
7.	Matnli faylda N ta talabaning familiyasi, ismi va kelgan viloyati yozilgan. Ushbu ma'lumotlarni to'plamga o'zlashtirib, talabalari eng kam kelgan viloyatni aniqlaydigan dastur tuzing.
8.	Matnli faylda N ta talabaning familiyasi, ismi va oilaviy ahvoli yozilgan. Ushbu ma'lumotlarni to'plamga o'zlashtirib, ota onasi yo'qlarini aniqlaydigan dastur tuzing.
9.	Matnli faylda N ta talabaning familiyasi, ismi va oilaviy ahvoli yozilgan. Ushbu ma'lumotlarni to'plamga o'zlashtirib, oilada yo'lg'iz o'g'il bo'lganlarini aniqlaydigan dastur tuzing.
10.	Matnli faylda N ta talabaning familiyasi, ismi va oilaviy ahvoli yozilgan. Ushbu ma'lumotlarni to'plamga o'zlashtirib, turmushga chiqqanlarni aniqlaydigan dastur tuzing.
11.	Matnli faylda N ta talabaning familiyasi, ismi va oilaviy ahvoli yozilgan. Ushbu ma'lumotlarni to'plamga o'zlashtirib, oilalali-Uylanganlarini aniqlaydigan dastur tuzing.
12.	Matnli faylda N ta talabaning familiyasi, ismi va oilaviy ahvoli yozilgan. Ushbu ma'lumotlarni to'plamga o'zlashtirib, TTJ dan yashaydiganlarini aniqlaydigan dastur tuzing.
13.	Matnli faylda N ta talabaning familiyasi, ismi va oilaviy ahvoli yozilgan. Ushbu ma'lumotlarni to'plamga o'zlashtirib, kvartirada yashaydiganlarini aniqlaydigan dastur tuzing.
14.	Matnli faylda N ta talabaning familiyasi, ismi va oilaviy ahvoli yozilgan. Ushbu ma'lumotlarni to'plamga o'zlashtirib, Dasturlash fanidan Olinpiada chempini bo'lganlarini aniqlaydigan dastur tuzing.
15.	Matnli faylda N ta talabaning familiyasi, ismi va oilaviy ahvoli yozilgan. Ushbu ma'lumotlarni to'plamga o'zlashtirib, Stependiant talabalarni aniqlaydigan dastur tuzing.

16.	Matnli faylda N ta talabning familiyasi, ismi va oilaviy ahvoli yozilgan. Ushbu ma'lumotlarni to'plamga o'zlashtirib, imkoniyati cheklanganlarni aniqlaydigan dastur tuzing.
-----	--

Takrorlash uchun savol va topshiriqlar:

1. Konteyner nima va uning maqsadi nimadan iborat?
2. Konteynerda asosan nechta bo'limlar mavjud?
3. Konteyner strukturasi?
4. Array sinfining nechta metodi bor? Konstruktorlarini yozing.
5. toString() metodining konstruktori va misol?
6. Shablon nima va uning maqsadi?
7. Metodlar uchun shablon qayerda va qanday ishlatiladi?
8. Konstruktorlar uchun shablon qayerda va qanday ishlatiladi?
9. Sinfklar uchun shablon qayerda va qanday ishlatiladi?
10. LinkedList sinfining nechta metodi bor? Konstruktorlarini yozing.
11. Tartiblangan konteynerlarga qanday sinflar kiradi?
12. Tartiblanmagan konteynerlarga qanday sinflar kiradi?
13. Konteyner nimadan tashkil topadi?
14. Konteyner dinamik xotiradan qanday foydalanadi?
15. Konteyner nima yordamida yaratiladi?
16. Standart algoritmlarni maqsadi nima?
17. Standart algoritmlar qanday vazifalarni yechishda qo'llaniladi?
18. O'zingizni konteyneringizni yarating.
19. Shaxsiy konteynerning maqsadi va vazilarini aniq belgilang.
20. Shaxsiy konteynerning yaratishga talablar ishlab chiqing.

3- LABORATORIYA ISHI

3. SONLI SINFLAR BILAN ISHLASH

3.1. Complex sonlar va ular bilan ishlashga doir amaliy dasturlar yaratish

Complex son deb $a+bi$ ifodaga aytiladi, bu yerda a va b haqiqiy sonlar, i – mavhum birlik bo‘lib, u yoki $i^2 = -1$ tengliklar bilan aniqlanadi; a – kompleks sonning haqiqiy qismi, bi – mavhum qismi deyiladi. Faqat mavhum qismining ishorasi bilan farq qiladigan ikki kompleks son: $a+bi$ va $a-bi$ o‘zaro qo‘shma deyiladi. Ko‘pincha $a+bi$ kompleks son bitta α harfi bilan belgilanadi: $\alpha = a+bi$. $a+bi$ kompleks sonning haqiqiy qismi $a = \operatorname{Re}\alpha$ bilan, mavhum qismining koeffitsientini $b = \operatorname{Im}\alpha$ bilan belgilaydilar. α kompleks sonning $a+bi$ ko‘rinishidagi yozuviga uning algebraik shakli deyiladi.

Agar ikkita $\alpha_1 = a_1 + b_1i$ va $\alpha_2 = a_2 + b_2i$ kompleks sonda $a_1 = a_2$, $b_1 = b_2$ bu ikki son teng deyiladi ($\alpha_1 = \alpha_2$). Agar $\alpha = a+bi$ kompleks sonda $a=0$, $b=0$ bo‘lsa, bu kompleks son 0 ga ($\alpha=0$) teng bo‘ladi. Agar $\alpha = a+bi$ kompleks sonda $b=0$ bo‘lsa, haqiqiy son hosil bo‘ladi; agar $a=0$ bo‘lsa, $0+bi=bi$ sof mavhum son deyiladi.

Complex sonlar sinfi quyidagi funksiyalardan tashkil topgan:

[abs](#), [acos](#), [acosh](#), [arg](#), [asin](#), [asinh](#), [atan](#), [atanh](#), [conj](#), [cos](#), [cosh](#), [exp](#),

[imag](#) - Murakkab sonning noma‘lum tarkibiy qismini ajratib oladi.

[log](#), [log10](#), [norm](#), [polar](#) - Belgilangan modul va argumentga mos keladigan murakkab sonni Cartesian shaklida qaytaradi.

[pow](#)- Murakkab son bo‘lgan bazani boshqa murakkab raqamning kuchiga ko‘tarish orqali olingan murakkab sonni baholaydi.

[proj](#), [real](#)- Murakkab sonning haqiqiy tarkibiy qismini ajratib oladi.

[sin](#), [sinh](#)- Kompleks sonning giperbolik sinusini qaytaradi.

[sqrt](#), [tan](#), [tanh](#) - Returns the hyperbolic tangent of a complex number.

Complex sonlar uchun quyidagi operatorlar qayta yuklangan:

operator!=	Haqiqiy va xayoliy qismlar uchun turlarning pastki qismiga tegishli bo'lishi mumkin bo'lgan ikkita yoki murakkab ikkala murakkab sonlar o'rtasidagi tengsizlik uchun testlar.
operator*	Haqiqiy va xayoliy qismlar uchun turlarning pastki qismiga tegishli bo'lishi mumkin bo'lgan ikkita murakkab raqamni ko'paytiradi.
operator+	Haqiqiy va xayoliy qismlar uchun tipning pastki qismiga tegishli bo'lishi mumkin bo'lgan ikkita murakkab sonni qo'shadi.
operator-	Haqiqiy va xayoliy qismlar uchun tipning pastki qismiga tegishli bo'lishi mumkin bo'lgan ikkita murakkab raqamni ayiradi.
operator/	Haqiqiy va xayoliy qismlar uchun turlarning pastki qismiga tegishli bo'lishi mumkin bo'lgan ikkita murakkab raqamni bo'ladi.
operator<<	Chiqish oqimiga murakkab raqamni kiritadigan shablon funktsiyasi.
operator==	Haqiqiy va xayoliy qismlar uchun turlarning pastki qismiga tegishli bo'lishi mumkin bo'lgan ikkita yoki murakkab ikkala murakkab sonlar o'rtasidagi tenglik uchun testlar.
operator>>	Kirish oqimidan murakkab qiymatni chiqaradigan shablon funktsiyasi.

Complex sonlar uchun quyidagi sinflar mavjud:

<u>complex<double></u>	Aniq ixtisoslashtirilgan sinf shablonlari ikki tomonlama tipdagi buyurtma qilingan juftlikni saqlaydigan ob'ektni tasvirlaydi, bu erda birinchisi murakkab sonning haqiqiy qismini, ikkinchisi esa xayoliy qismni anglatadi.
<u>complex<float></u>	Shubhasiz ixtisoslashtirilgan sinf shablonlari buyurtma qilingan juftlikni saqlaydigan ob'ektni tavsiflaydi, har ikkala turdagi float ham, birinchisi murakkab sonning haqiqiy qismini, ikkinchisi esa xayoliy qismni anglatadi.
<u>complex<long double></u>	Aniq ixtisoslashtirilgan shablon shablonni buyurtma qilingan juftlikni saqlaydigan ob'ektni tavsiflaydi, har ikkala turdagi uzun juft juftlik, bu erda murakkab raqamning haqiqiy qismi, ikkinchisi xayoliy qism tasvirlangan.
<u>complex</u>	Sinf shablonlari murakkab sonlar tizimini ifodalash va murakkab arifmetik amallarni bajarish uchun ishlatiladigan ob'ektni tavsiflaydi.

Kompleks sonlarning yuqori keltirilgan tarkibiy qismlarini bazi birlariga misollar keltiramiz.

1. **abs()** complex funksiyasiga misol:

Dastur matni:

```

1. #include "stdafx.h"
2. #include <complex>
3. #include <iostream>
4. using namespace System;
5. using namespace std;
6. int main( ) {
7. using namespace std;
8. double pi = 3.14159265359;
9. complex <double> c1 ( polar ( 5.0 ) ); // Default
   argument = 0
10. complex <double> c2 ( polar ( 5.0 , pi / 6 ) );
11. complex <double> c3 ( polar ( 5.0 , 13 * pi / 6 ) );
12. cout << "c1 = polar ( 5.0 ) = " << c1 << endl;
13. cout<< "c2 = polar ( 5.0 , pi / 6 ) = " << c2 << endl;
14. cout<<"c3 = polar (5.0, 13*pi / 6 ) = " << c3 << endl;
15. double absc1 = abs ( c1 );
16. double argc1 = arg ( c1 );
17. cout << "C1 moduli c1 dan quyidagilar yordamida
   tiklanadi: abs ( c1 ) = "
18. << absc1 << endl;
19. cout << "C1 argumenti c1 dan quyidagilar yordamida
   tiklanadi:\n arg ( c1 ) = "
20. << argc1 << " radians, qaysiki " << argc1 * 180 / pi
21. << " darajasi." << endl;
22. double absc2 = abs ( c2 );
23. double argc2 = arg ( c2 );
24. cout << "C2 moduli c2 dan s2 yordamida tiklanadi: abs
   ( c2 ) = "
25. << absc2 << endl;
26. cout << "C2 argumenti c2 dan quyidagilar yordamida
   tiklanadi:\n arg ( c2 ) = "
27. << argc2 << " radianda, qaysiki " << argc2 * 180 / pi
28. << " darajalar." << endl;
29. if ( (arg ( c2 ) <= ( arg ( c3 ) + .00000001) ) ||
30. (arg ( c2 ) >= ( arg ( c3 ) - .00000001) ) )
31. cout << " c2 & c3 kompleks sonlar bir xil
   argumentlardan "
32. << "tashkil topgan."<< endl;
33. else
34. cout << " c2 & c3 kompleks sonlar bir xil
   argumentlardan "
35. << "tashkil topmagan."<< endl;

```

```
36. getch();
37. }
```

Dastur natijasi:

$c1 = \text{polar} (5.0) = (5,0)$

$c2 = \text{polar} (5.0 , \pi / 6) = (4.33013,2.5)$

$c3 = \text{polar} (5.0 , 13 * \pi / 6) = (4.33013,2.5)$

C1 moduli c1 dan quyidagilar yordamida tiklanadi: $\text{abs} (c1) = 5$

C1 argumenti c1 dan quyidagilar yordamida tiklanadi:

$\text{arg} (c1) = 0$ radians, qaysiki 0 darajasi.

C2 moduli c2 dan s2 yordamida tiklanadi: $\text{abs} (c2) = 5$

C2 argumenti c2 dan quyidagilar yordamida tiklanadi:

$\text{arg} (c2) = 0.523599$ radians, qaysiki 30 darajalar.

c2 & c3 kompleks sonlar bir xil argumentlardan tashkil topgan.

2. **comj()** complex funksiyasiga misol:

Dastur matni:

```
1. #include "stdafx.h"
2. #include <complex>
3. #include <iostream>
4. int main( ) {
5. using namespace std;
6. complex <double> c1 ( 4.0 , 3.0 );
7. cout << "Kompleks raqam c1 = " << c1 << endl;
8. double dr1 = real ( c1 );
9. cout << " c1 ning haqiqiy qismi bu haqiqiy ( c1 ) = "
10. << dr1 << "." << endl;
11. double di1 = imag ( c1 );
12. cout << " c1 ning no'malum qismi bu imag ( c1 ) = "
13. << di1 << "." << endl;
14. complex <double> c2 = conj ( c1 );
15. cout<<"c1 ning kompleks kasr qismi bu c2 = conj(c1)= "
16. << c2 << endl;
17. double dr2 = real ( c2 );
18. cout << " c2 ning haqiqiy qismi bu real ( c2 ) = "
19. << dr2 << "." << endl;
```

```

20. double di2 = imag ( c2 );
21. cout << " c2 ning noʻmalum qismi bu imag ( c2 ) = "
22. << di2 << "." << endl;
23. complex <double> c3 = c1 * c2;
24. cout<<"(c1 * conj (c1) ) ning haqiqiysi bu c1 * c2 = "
25. << real( c3 ) << endl;
26. getchchar();
27.}

```

Dastur natijasi:

Kompleks raqam $c1 = (4,3)$

$c1$ ning haqiqiy qismi bu haqiqiy ($c1$) = 4.

$c1$ ning noʻmalum qismi bu imag ($c1$) = 3.

$c1$ ning kompleks kasr qismi bu $c2 = \text{conj} (c1) = (4,-3)$

$c2$ ning haqiqiy qismi bu real ($c2$) = 4.

$c2$ ning noʻmalum qismi bu imag ($c2$) = -3.

($c1 * \text{conj} (c1)$) ning haqiqiysi bu $c1 * c2 = 25$

3. **imag()** complex funksiyasiga misol:

Dastur matni:

```

1. #include "stdafx.h"
2. #include <complex>
3. #include <iostream>
4. int main( ){
5. using namespace std;
6. complex <double> c1 ( 4.0 , 3.0 );
7. cout << " c1 kompleks son = " << c1 << endl;
8. double dr1 = real ( c1 );
9. cout << " c1 ning haqiqiy qismi bu real ( c1 ) = "
10. << dr1 << "." << endl;
11. double di1 = imag ( c1 );
12. cout << " c1 ning nomaʻlum qismi bu imag ( c1 ) = "
13. << di1 << "." << endl;
14. getchchar();
15. }

```

Dastur natijasi:

$c1$ kompleks son = (4,3)

$c1$ ning haqiqiy qismi bu $\text{real}(c1) = 4$.

$c1$ ning noma'lum qismi bu $\text{imag}(c1) = 3$.

4. **polar()** complex funksiyasiga misol:

Dastur matni:

```
1. #include "stdafx.h"
2. #include <complex>
3. #include <iostream>
4. int main( ){
5. using namespace std;
6. double pi = 3.14159265359;
7. complex <double> c1 ( polar ( 5.0 ) );
8. // Default argument = 0
9. complex <double> c2 ( polar ( 5.0 , pi / 6 ) );
10. complex <double> c3 ( polar ( 5.0 , 13 * pi / 6 ) );
11. cout << "c1 = polar ( 5.0 ) = " << c1 << endl;
12. cout << "c2= polar ( 5.0 , pi / 6 ) = " << c2 << endl;
13. cout << "c3= polar(5.0, 13*pi / 6 ) = " << c3 << endl;
14. if ( (arg ( c2 ) <= ( arg ( c3 ) + .00000001) ) ||
15. (arg ( c2 ) >= ( arg ( c3 ) - .00000001) ) )
16. cout << " c2 & c3 kompleks sonlar bir xil
    argumentlardan "<< "tashkil topgan."<< endl;
17. else
18. cout << " c2 & c3 kompleks sonlar bir xil
    argumentlardan "<< "tashkil topmagan." << endl;
19. double absc2 = abs ( c2 );
20. double argc2 = arg ( c2 );
21. cout << " c2 ning moduli bu : abs ( c2 ) = "
22. << absc2 << endl;
23. cout << " c2 argumenti bu :\n arg ( c2 ) = "
24. << argc2 << " radianda, qaysiki " << argc2 * 180 / pi
25. << " daraja." << endl;
26. getchar();
27. }
```

Dastur natijasi:

$c1 = \text{polar}(5.0) = (5,0)$

$c2 = \text{polar} (5.0 , \pi / 6) = (4.33013, 2.5)$

$c3 = \text{polar} (5.0 , 13 * \pi / 6) = (4.33013, 2.5)$

$c2$ & $c3$ kompleks sonlar bir xil argumentlardan tashkil topgan.

$c2$ ning moduli bu : $\text{abs} (c2) = 5$

$c2$ argumenti bu :

$\text{arg} (c2) = 0.523599$ radianda, qaysiki 30 daraja.

Kompleks sonlar uchun qayta yuklangan teng emas operatori - **operator!=** ga misol:

Dastur matni:

```
1. #include "stdafx.h"
2. #include <complex>
3. #include <iostream>
4. int main( ){
5. using namespace std;
6. double pi = 3.14159265359;
7. complex <double> c11 ( polar (3.0, pi / 6 ) );
8. complex <double> cr1a ( polar (3.0, pi /6 ) );
9. complex <double> cr1b ( polar (2.0, pi / 3 ) );
10. cout << " c11 kompleks sonning chap tarafi= " << c11 << endl;
11. cout<<"cr1a kompleks sonning birinchi o'ng tarafi = "
    <<cr1a<<endl;
12. cout<<"cr1b kompleks sonning ikkinchi o'ng tarafi = "
    <<cr1b<<endl;
13. if ( c11 != cr1a )
14. cout << " c11 & cr1a kompleks sonlar teng emas." << endl;
15. else
16. cout << " c11 & cr1a kompleks sonlar teng." <<endl;
17. if ( c11 != cr1b )
18. cout << " c11 & cr1b kompleks sonlar teng emas."<< endl;
19. else
20. cout << " c11 & cr1b kompleks sonlar teng." << endl;
21. cout << endl;
22. complex <int> c12a ( 3, 4 );
23. complex <int> c12b ( 5,0 );
24. int cr2a =3;
25. int cr2b =5;
26. cout<<"c12a kompleks sonning birinchi chap tarafi ="<<c12a <<
    endl;
27. cout<<"cr2a kompleks sonning birinchi o'ng tarafi = " << cr2a <<
    endl;
28. if ( c12a != cr2a )
29. cout << " c12a & cr2a kompleks sonlar teng emas." << endl;
30. else
```

```

31. cout << " c12a & cr2a kompleks sonlar teng." << endl;
32. cout<<"c12b kompleks sonning ikkinchi chap tarafi="<<c12b<< endl;
33. cout<<"cr2b kompleks sonning ikkinchi o'ng tarafi="<<cr2b<< endl;
34. if ( c12b != cr2b )
35. cout << " c12b & cr2b kompleks sonlar teng emas." << endl;
36. else
37. cout << " c12b & cr2b kompleks sonlar teng." << endl;
38. cout << endl;
39. double c13a =3;
40. double c13b =5;
41. complex <double> cr3a ( 3, 4 );
42. complex <double> cr3b ( 5,0 );
43. cout<<"c13a kompleks sonning birinchi chap tarafi="<<c13a<< endl;
44. cout<<"cr3a kompleks sonning birinchi o'ng tarafi="<<cr3a<< endl;
45. if ( c13a != cr3a )
46. cout << " c13a & cr3a kompleks sonlar teng emas." << endl;
47. else
48. cout << " c13a & cr3a kompleks sonlar teng." << endl;
49. cout<<"c13b kompleks sonning ikkinchi chap tarafi="<<c13b<<endl;
50. cout<<"cr3b kompleks sonning ikkinchi o'ng tarafi="<<cr3b<<endl;
51. if ( c13b != cr3b )
52. cout << " c13b & cr3b kompleks sonlar teng emas." << endl;
53. else
54. cout << " c13b & cr3b kompleks sonlar teng." << endl;
55. cout << endl;
56. getchar();
57. }

```

Dastur natijasi:

c11 kompleks sonning chap tarafi= (2.59808,1.5)

cr1a kompleks sonning birinchi o'ng tarafi = (2.59808,1.5)

cr1b kompleks sonning ikkinchi o'ng tarafi = (1,1.73205)

c11 & cr1a kompleks sonlar teng.

c11 & cr1b kompleks sonlar teng emas.

c12a kompleks sonning birinchi chap tarafi = (3,4)

cr2a kompleks sonning birinchi o'ng tarafi = 3

c12a & cr2a kompleks sonlar teng emas.

c12b kompleks sonning ikkinchi chap tarafi = (5,0)

cr2b kompleks sonning ikkinchi o'ng tarafi = 5

cl2b & cr2b kompleks sonlar teng.

cl3a kompleks sonning birinchi chap tarafi = 3

cr3a kompleks sonning birinchi o'ng tarafi = (3,4)

cl3a & cr3a kompleks sonlar teng emas.

cl3b kompleks sonning ikkinchi chap tarafi = 5

cr3b kompleks sonning ikkinchi o'ng tarafi = (5,0)

cl3b & cr3b kompleks sonlar teng.

Laboratoriya mashg'ulotini bajarish bo'yicha variantlar

Variandda berilgan vazifalar complex sonlar yordamida amalga oshirilishi kerak.

1. N ta complex son berilgan. Ularning mavhum qismlari tasodifiy toq sonlar bilan to'ldirilsin hamda ushbu N ta complex sonning yig'indisi topilsin.
2. N ta complex son berilgan. Ularning real qismlari tasodifiy juft sonlar bilan to'ldirilsin hamda ushbu N ta complex sonning toq indexli elementlarining yig'indisi topilsin.
3. N ta complex son berilgan. Ularning real va mavhum qismlari (complex() yordamida) tasodifiy bilan to'ldirilsin hamda ushbu N ta complex sonning tublarini yig'indisi topilsin.
4. N ta complex son berilgan. Ularning real qismlari tasodifiy juft sonlar bilan, mavhum qismlari tasodifiy toq sonlar bilan to'ldirilsin hamda ushbu N ta complex sonlarning M sonidan kattalarini yig'indisi topilsin.
5. N ta complex son berilgan. Ularning real qismlari tasodifiy tub sonlar bilan, mavhum qismlari 1 dan N gacha sonlar bilan to'ldirilsin hamda ushbu N ta complex sonlarning M soniga karralilarining yig'indisi topilsin.
6. N ta complex son berilgan. Ularning real qismlari tasodifiy takrorlanmas sonlar bilan, mavhum qismlari 1 dan N gacha manfiy sonlar bilan to'ldirilsin hamda ushbu N ta complex sonlarni o'sish tartibida saralab ekranga chiqarilsin.
7. N ta complex son berilgan. Ularning real qismlari tasodifiy takrorlanmas sonlar bilan, mavhum qismlari 1 dan N gacha manfiy sonlar bilan to'ldirilsin hamda ushbu N ta complex sonlarni jadval ko'rinishida har bir satrini o'sish tartibida saralab ekranga chiqarilsin.
8. N ta complex son berilgan. Ularning real qismlari tasodifiy takrorlanmas sonlar bilan, mavhum qismlari 1 dan N gacha manfiy toq sonlar bilan

- to'ldirilsin hamda ushbu N ta complex sonlarni va ularning mavhum qismlaridan C massivni, real qismlaridan esa D massivni hosil qilib, ekranga chiqarilsin.
9. N ta complex son berilgan. Ularning real qismlari tasodifiy sonlar bilan, mavhum qismlari 1 dan N gacha sonlar bilan to'ldirilsin hamda ushbu N ta complex sonlarni va ularning mavhum qismlari bilan real qismlarini yig'indilarini ekranga chiqarilsin.
 10. N ta complex son berilgan. Ularning real qismlari tasodifiy sonlar bilan, mavhum qismlari 1 dan N gacha sonlar bilan to'ldirilsin hamda ushbu N ta complex sonlarni va ularning mavhum qismlarida takrorlangan sonlarni va ularning soni ekranga chiqarilsin.
 11. N ta complex son berilgan. Ularning real qismlari tasodifiy sonlar bilan, mavhum qismlari N ta ixtiyoriy sonlar bilan to'ldirilsin hamda ushbu N ta complex sonlarni va ularning real qismlarida 2 ta martadan ko'p takrorlangan sonlarni va ularning soni ekranga chiqarilsin.
 12. N ta complex son berilgan. Ularning real qismlari tasodifiy sonlar bilan, mavhum qismlari N ta ixtiyoriy sonlar bilan to'ldirilsin hamda ushbu N ta complex sonlarni va ikkala qismini yig'indisidan hosil bo'lgan sonlardan max va min larini va ular bittadan ko'p bo'lsa, ular ham ekranga chiqarilsin.
 13. N ta complex son berilgan. Ularning real qismlari tasodifiy sonlar bilan, mavhum qismlari N ta ixtiyoriy sonlar bilan to'ldirilsin hamda ushbu N ta complex sonlarni va ikkala qismini yig'indisidan hosil bo'lgan sonlarning raqamlar yig'indisi max va min larini va ular bittadan ko'p bo'lsa, ular ham ekranga chiqarilsin.
 14. N ta complex son berilgan. Ularning real qismlari tasodifiy sonlar bilan, mavhum qismlari N ta ixtiyoriy sonlar bilan to'ldirilsin hamda ushbu N ta complex sonlarni real qismining max mi, mavhum qismining max midan katta yoki kichik ekanligi aniqlansin va ular ekranga chiqarilsin.
 15. N ta complex son berilgan. Ularning real qismlari tasodifiy sonlar bilan, mavhum qismlari N ta ixtiyoriy sonlar bilan to'ldirilsin hamda ushbu N ta complex sonlarni ikkala qismida 2 martadan ko'p takrorlanganlarini va ularni yig'indisi ekranga chiqarilsin.
 16. N ta complex son berilgan. Ularning real qismlari tasodifiy sonlar bilan, mavhum qismlari N ta ixtiyoriy sonlar bilan to'ldirilsin hamda ushbu N ta complex sonlarni, real qismidagi sonlar mavhum qismida necha merta qatnashganligi aniqlansin va ekranga chiqarilsin.
 17. N ta complex son berilgan. Ularning real qismlari tasodifiy sonlar bilan, mavhum qismlari N ta 2 ning N darajasi (2^N) ga karrali bo'lgan sonlar bilan to'ldirilsin hamda ushbu N ta complex sonlarni jadval ko'rinishida ekranga chiqarilsin.

18. N ta complex son berilgan. Ularning real qismlari tasodifiy sonlar bilan, mavhum qismlari N ta murakkab ($6=1+2+3$ – bo‘luvchilari yig‘indisiga teng bo‘lgan son) sonlar bilan to‘ldirilsin hamda ushbu N ta complex sonlarni jadval ko‘rinishida ekranga chiqarilsin.
19. N ta complex son berilgan. Ularning real qismlari tasodifiy sonlar bilan, mavhum qismlari real qismidagi sonning kubiga o‘zgartirilsin hamda ushbu N ta complex sonlarni real qismini alohida, mavhum qismini alohida ko‘rinishida ekranga chiqarilsin.
20. N ta complex son berilgan. Ularning real qismlari matnli fayldagi sonlar bilan, mavhum qismlari binary fayldagi sonlar bilan to‘ldirilsin hamda ushbu N ta complex sonlarni real qismini bilan mavhum qismini yig‘indisi ekranga chiqarilsin.

Takrorlash uchun savol va topshiriqlar:

1. Complex sonlar nima?
2. Complex son nechta qismdan iborat?
3. Complex sonlar bilan ishlovchi funksiyalarni bilasizmi?
4. Complex sonlar bilan ishlovchi fuksiyalar tarkibi qanday tuzilgan?
5. Complex sonlar qanday vazifalarni yechishda foydalaniladi?
6. Complex sonlarning real qismiga qanday murojaat qilinadi?
7. Complex sonlarning mavhum qismlariga qanday murojat qilinadi?
8. Complex sonlar bilan ishlovchi sinflarni aniqlang.
9. Complex sonlar uchun operatorlarni qayta yuklash kerakmi?
10. Complex sonlardaf foydalanish talab etiladigan vazifalarni aniqlang.

4- LABORATORIYA ISHI

SINTAKSIS TAHLIL VA MURAKKAB SARALASH ALGORITMLARI

4. IFODALARNING SINTAKSIS TAHLILI

4.1. Leksik tahlilni amalga oshiruvchi amaliy dastur tuzish

Ishning maqsadi: C++ dasturlash tilida **leksik va sintaksis tahlil** qilish va uning dasturini yaratish ko'nikmalarini egallash.

Masalaning qo'yilishi:

Matnli faylda berilgan ma'lumotlarni **leksik va sintaksis tahlil** qilish.

Masala: Matnli faylda “ ; ” (nuqta-vergul) belgisi bilan ajratilgan **arifmetik iboralar** mavjud. Arifmetik ifodalar: **identifikatorlardan, o'nlik suzuvchi nuqta raqamlaridan**, o'zlashtirish belgisi (=), ishoralar +, -, *, / va figurali qavs({}) lardan iborat.

1) Topshiriqqa muvofiq kirish matnini **leksik** tahlilini bajaradigan va ularning turlarini ko'rsatgan holda **tokenlar jadvalini** tuzadigan dastur yozing. Dastur kirish matnida **leksik tahlil** bosqichida aniqlanishi mumkin bo'lgan xatolar (**sintaksis tahlil**) mavjudligi to'g'risida xabarlarini chiqarishi kerak;

2) **leksik va sintaksis tahlil** analizator kodini yaratishda Visual C++ dan kompilyator sifatida foydalaning.

Masalani yechish g'oyasi:

Oqimlar sinfi (**fstream.h**) dan foydalanib, matnli fayl ma'lumotlari o'zlashtiriladi. **Tokenlar jadvali** yaratish uchun birinchi: faylning nechta satr borligi **getline(oqi, line)** funksiyasidan foydalanilaniq aniqlanadi. Xar bir satr sikl qadamlar bilan sanaladi (**i++**). O'zlashtirib olingan satr o'zgaruvchi (**line**) ning har - bir elementi vazifada belgilangan belgilarga tekshiriladi.

Izoh: **Leksik va sintaksis tahlil** masalani to'liq yechish uchun ko'p vaqt sarflanishi inobatga olingan holda har - bir talabaga leksik tahlil bosqichida alohida belgilar ajratiladi. Sintaksis tahlil da esa ushbu aniqlangan belgilar ketma-ketligi

tekshiriladi. Agar ketma-ketliklar noto'g'ri joylashgan bo'lsa, ushbu xatolik haqida xabar beruvchi dastur tuzilishi kerak.

Leksik va sintaksis tahlil haqida

Kompilyatorlar va ularning turlari:

Bir tilda yozilgan dasturni protsessorning boshqa tilda ifodalangan dasturiga aylantiruvchi dasturga kompilyator deyiladi.

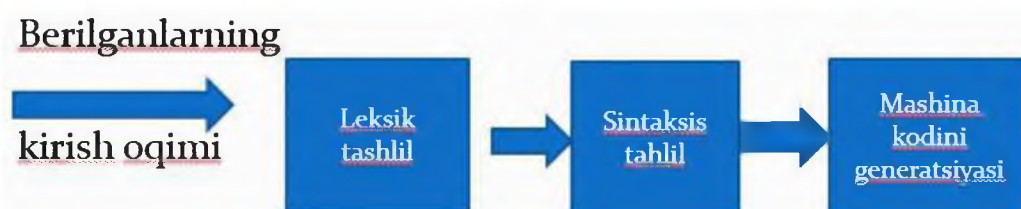
Masalan, kompilyator C tilida yozilgan dasturni olib, uni **Assembler** tilida yozilgan dasturga aylantiradi.

Ko'pincha kompilyator va translyator tushunchalari bir ma'noda ishlatiladi. Shuning uchun ham **C**, **C++**, **Paskal kompilyatorlari** degan tushunchalari ishlatiladi.

Ixtiyoriy kompilyator tarkibi quyidagi 3ta qismdan iborat:

1. **Leksik** tahlil.
2. **Sintaksis** tahlil.
3. **Mashina kodini generatsiya** qilish.

Kompilyatorlarning o'zaro bog'lanish tartibi quyidagicha:



Bu bloklarning xar biriga alohida dasturlar mos keladi.

C++ tili va uning leksik asosi

1. C++ tilidagi programma tuzilishi va uning kompilyatsiyasi.
2. C++tili alfaviti va leksemalar.
3. Identifikatorlar va kalit so'zlar.

Tayanch iboralar: Kompilyatsiya, fayl kengaytmasi, direktivalar, leksemalar, izohlar, oqimli o'qish va yozish funksiyalari, til alfaviti, identifikatorlar, kalit so'zlar.

C++ tilidagi programma tuzilishi va uning kompilyatsiyasi:[2(11-13), 3(7-

9)]C++ tilida programma yaratish bir nechta bosqichlardan iborat bo‘ladi. Dastlab, matn tahririda (odatda programmalash muhitining tahririda) programma matni teriladi, bu faylning kengaytmasi «.cpp» bo‘ladi. Keyingi bosqichda programma matni yozilgan fayl kompilyatorga uzatiladi, agarda programmada xatoliklar bo‘lmasa, kompilyator «.obj» kengaytmali obyekt modul faylini hosil qiladi. Oxirgi qadamda komponovka (yig‘uvchi) yordamida «.exe» kengaytmali bajariluvchi fayl programma hosil bo‘ladi. Bosqichlarda yuzaga keluvchi fayllarning nomlari boshlang‘ich matn faylining nomi bilan bir xil bo‘ladi.

Kompilyatsiya jarayonining o‘zi ham ikkita bosqichdan tashkil topadi. Boshida preprotessor ishlaydi, u matndagi kompilyatsiya direktivalarini bajaradi, xususan #include direktivasi bo‘yicha ko‘rsatilgan kutubxonalaridan C++ tilida yozilgan modullarni programma tarkibiga kiritadi. Shundan so‘ng kengaytirilgan programma matni kompilyatorga uzatiladi. Kompilyator o‘zi ham programma bo‘lib, uning uchun kiruvchi ma’lumot bo‘lib, C++ tilida yozilgan programma matni hisoblanadi. Kompilyator programma matnini leksema (atomar) elementlarga ajratadi va uni leksik, keyinchalik sintaktik tahlil qiladi. Leksik tahlil jarayonida u matnni leksemalarga ajratish uchun «probel ajratuvchisini» ishlatadi. Probel ajratuvchisiga - probel belgisi (‘—‘), ‘\t‘ - tabulyatsiya belgisi, ‘\n‘ - keyingi qatorga o‘tish belgisi, boshqa ajratuvchilar va izohlar (kommentariylar) kiradi.

Programma matni tushunarli bo‘lishi uchun izohlar ishlatiladi. Izohlar kompilyator tomonidan «o‘tkazib» yuboriladi va ular programma amal qilishiga hech qanday ta‘sir qilmaydi. C++ tilida izohlar ikki ko‘rinishda yozilishi mumkin.

Birinchisida "/*" dan boshlanib, "*/" belgalari bilan tugagan barcha belgilar ketma-ketligi izoh hisoblanadi, ikkinchisi «satriy izoh» deb nomlanadi va u "///" belgilardan boshlangan va satr oxirigacha yozilgan belgilar ketma-ketligi bo‘ladi. Izohning birinchi ko‘rinishida yozilgan izohlar bir necha satr bo‘lishi va ulardan keyin C++ operatorlari davom etishi mumkin.

Misol.

```
int main(){
```

```
// bu qator izoh hisoblanadi
```

```
int a=0; //int d;
```

```
int c;
```

```
/* int b=15 */
```

```
/*- izoh boshlanishi
```

```
a=c;
```

```
izoh tugashi */
```

```
return 0; }
```

Programmada **d**, **b** o'zgaruvchilar e'lonlari inobatga olinmaydi va **a=c** amali bajarilmaydi.

Quyida C++ tilidagi sodda dastur matni keltirilgan.

```
# include // sarlavha faylni qo'shish
```

```
int main () // bosh funksiya tavsifi
```

```
{ // blok boshlanishi
```

```
cout << "Salom Olam! \n"; // satrni chop etish
```

```
return 0; // funksiya qaytaradigan qiymat
```

```
} // blok tugashi
```

Programma bajarilishi natijasida ekranga "**Salom Olam!**" satri chop etiladi. Programmaning 1-satrida `#include..` preprotssessor direktivasi bo'lib, programma kodiga oqimli o'qish/yo'zish funksiyalari va uning o'zgaruvchilari e'loni joylashgan «`iostream.h`» sarlavha faylini qo'shadi. Keyingi qatorlarda programmaning yagona, asosiy funksiyasi -`main()` funksiyasi tavsifi keltirilgan. Shuni qayd etish kerakki, C++ programmasida albatta `main()` funksiyasi bo'lishi shart va programma shu funksiyani bajarish bilan o'z ishini boshlaydi. Programma tanasida konsol rejimida belgilar ketma-ketligini oqimga chiqarish amali qo'llanilgan. Ma'lumotlarni standart oqimga (ekranga) chiqarish uchun quyidagi format ishlatilgan:

```
cout << ;
```

Bu yerda sifatida o'zgaruvchi yoki sintaksisi to'g'ri yozilgan va qandaydir qiymat qabul qiluvchi til ifodasi kelishi mumkin (keyinchalik, burchak qavs ichiga olingan o'zbekcha satr ostini til tarkibiga kirmaydigan tushuncha deb qabul qilish kerak).

Masalan:

```
int uzg=324;  
cout<<
```

Berilganlarni standart oqimdan (odatda klaviaturadan) o'qish quyidagi formatda amalga oshiriladi:

```
cin>> ;
```

Bu yerda qiymat qabul qiluvchi o'zgaruvchining nomi.

Misol:

```
int Yosh;  
cout <<"Yoshingizni kiriting_";  
cin>>Yosh;
```

Butun turdagi Yosh o'zgaruvchisi kiritilgan qiymatni o'zlashtiradi. Kiritilgan qiymatni o'zgaruvchi turiga mos kelishini tekshirish mas'uliyati programma tuzuvchisining zimmasiga yuklanadi.

Bir paytning o'zida probel () vositasida bir nechta va har xil turdagi qiymatlarni oqimdan kiritish mumkin. Qiymat kiritish tugmasini bosish bilan tugaydi. Agar kiritilgan qiymatlar soni o'zgaruvchilar sonidan ko'p bo'lsa, «ortiqcha» qiymatlar bufer xotirada saqlanib qoladi.

```
# include  
int main (){  
int x,y;  
float z;  
cin>>x>>y>>z;  
cout << "O'qilgan qiymatlar \n";  
cout<<x'\t'<<y'\t'<<z;
```

return 0;

}

O'zgaruvchilarga qiymat kiritish uchun klaviatura orqali harakati amalga oshiriladi. Shuni qayd etish kerakki, oqimga qiymat kiritishda probel ajratuvchi hisoblanadi. Haqiqiy sonning butun va kasr qismlari ' . ' belgisi bilan ajratiladi.

C++ tili alfaviti va leksemalar:[1(35-43), 2(13-19), 3(9-10)] C ++ tili alfaviti va leksemalariga quyidagilar kiradi:

- katta va kichik lotin alfaviti harflari;

- raqamlar - 0,1,2,3,4,5,6,7,8,9;

- maxsus belgilar: " } | [] () + - / % \ ; ' : ? <=> _ ! & ~ # ^ . *

Alfavit belgilaridan tilning leksemalari shakllantiriladi:

- identifikatorlar;

- kalit (xizmatchi yoki zahiralangan) so'zlar;

- o'zgarmlar;

- amallar belgilanishlari;

- ajratuvchilar.

Identifikatorlar va kalit so'zlar:[1(43-52),3(10-11)]Programmash tilining muhim tayanch tushunchalaridan biri identifikator tushunchasidir. Identifikatordeganda katta va kichik lotin harflari, raqamlar va tag chiziq ('_') belgilaridan tashkil topgan va raqamdan boshlanmaydigan belgilar ketma-ketligi tushuniladi. Identifikatorlarda harflarning registrlari (katta yoki kichikligi) hisobga olinadi. Masalan, RUN, run, Run - bu har xil identifikatorlardir. Identifikator uzunligiga chegara qo'yilmagan, lekin ular kompilyator tomonidan faqat boshidagi 32 belgisi bilan farqlanadi. Identifikatorlar kalit so'zlar, o'zgaruvchilar, funksiyalar, nishonlar va boshqa obyektlarni nomlashda ishlatiladi. C++ tilining kalit so'zlariga quyidagilar kiradi:

asm, auto, break, case, catch, char, class, const, continue, default, delete, do, double, else, enum, explicit, extern, float, for, friend, goto, if, inline, int, long, mutable, new, operator, private, protected, public, register, return, short, signed, sizeof, static, struct, switch, template, this, throw, try, typedef, typename, union, unsigned, virtual, void, volatile, while.

Yuqorida keltirilgan identifikatorlarni boshqa maqsadda ishlatish mumkin emas. Prozessor registrlarini belgilash uchun quyidagi soʻzlar ishlatiladi: `_AH, _AL, _AX, _EAX, _BN, _BL, _BX, _EVBX, _CL, _CN, _CX, _ESX, _DN, _DL, _DX, _EDX, _CS, _ESR, EBP, _FS, __GS, _DI, _EDI, _SI, _ESI, __BP, SP, DS, _ES, SS, _FLAGS`.

Bulardan tashqari «`_`» (ikkita tag chiziq) belgilaridan boshlangan identifikatorlar kutubxonalar uchun zahiralangan. Shu sababli «`_`» va «`__`» belgilarni identifikatorning birinchi belgisi sifatida ishlatmagan maqul. Identifikator belgilar orasida probel ishlatish mumkin emas, zarur boʻlganda uning oʻrniga «`_`» ishlatish mumkin: `silindr_radiusi`, `aylana_diametri`.

Dastur matni:

```
1. #include "stdafx.h"
2. #include<iostream>
3. #include<fstream>
4. #include<string.h>
5. #include<string>
6. #include<stdlib.h>
7. using namespace std;
8. int main(){
9. // matnli faylni oʻqish uchun ochish
10.     ifstream oqi("1.cpp");
11.     if(!oqi.is_open()){
12.         cout<<"Fayl topilmadi"<<endl; getch(); exit(1);
13.     }
14.     int i=1;
15.     string line, str;
16.     bool s=0,s1=0, s2=0; int e1=0,e2=0;
17.     int z2=0;
18.     // fayl oʻhirigacha oʻqish
19.     while (getline(oqi, line)) // qatorning ohirigacha oʻqish
20.     {
21.         if(oqi.fail()) break; // faylda maʼlumotar tugashi
22.         e1=0; e2=0; string h=""; int h2=0;
```

```

23.     // fayldan o'zlashtirilgan qator ohirigacha o'qish
24.     // Ushbu satrda vazifada berilgan aktiv belgilarni aniqlash
25.     for(int t=0; t<line.length();t++){
26.         s=0; s1=0; s2=0; e1=0; e2=0;
27.         string d; d.assign(line,t,1); // line satrini har - bir
           elementini d ga o'zlashtirish
28.         // leksik tahlil uchun
29.         if(d=="=" || d=="+" || d=="-" || d=="*" || d=="/" ){
30.             // sintaksis tahlil uchun
31.             if(line[t-1]=='+' || line[t-1]=='-' || line[t-1]=='*' ||
           line[t-1]=='/'){
32.                 if(d!="=" && (line[t-2]!='+' || line[t-2]!='-' || line[t-
           2]!='*' || line[t-2]!='/') ){
33.                     cout<<endl<<"Arifmetik ifodada "<<line[t-1]<<" ishora ketma-
           ket kelgan"<<endl;
34.                     e1=1;
35.                 }
36.             }
37.             if(line[t+1]=='+' || line[t+1]=='-' || line[t+1]=='*' ||
           line[t+1]=='/'){
38.                 // sintaksis tahlil uchun
39.                 if(d=="="){
40.                     cout<<endl<<"Barobardan keyin ishora mumkin emas"<<endl;
41.                     e1=1;
42.                 }
43.             }

44.         float a;
45.         // raqam ekanligini aniqlash
46.         if(atof(h.c_str())){ a=atof(h.c_str());
47.             cout<<"NUMBER("<<i<<","<<t-h2+1<<") - "<<a<<endl;
48.         }
49.         else if(e1!=1){
50.             cout<<"IDENTIFIKATOR("<<i<<","<<t-h2+1<<") - "<<h<<endl;

51.             cout<<"\t\t\t\t OPERATOR("<<i<<","<<t+1<<") - "<<d<<endl;
           s=1; s1=0; s2=0;
52.         }
53.         h2=0; h.clear();
54.         }
55.         else if(d==";" || d==","){ e2=t; s1=1; s=0; s2=0;
56.             // sintksis tahlil uchun
57.             if((d!=",") && line[t-1]=='+' || line[t-1]=='-' || line[t-
           1]=='*' || line[t-1]=='/'){
58.                 if(d!="=" && (line[t-2]=='+' || line[t-2]=='-' || line[t-
           2]=='*' || line[t-2]=='/') ){
59.                     cout<<endl<<"Arifmetik ifodada "<<line[t-1]<<" ishora ketma-
           ket kelgan"<<endl;
60.                     e1=1;

```

```

61.     }
62.     }
63.     if(line[t+1]=='+' || line[t+1]=='-' || line[t+1]=='*' ||
line[t+1]=='/'){
64.         // sintksis tahlil uchun
65.         if(d==","){
66.             cout<<endl<<"Verguldan keyin ishora mumkin emas"<<endl;
67.             e1=1;
68.         }
69.     }
70.     // raqam ekanligini aniqlash
71.     float a;
72.     if(atof(h.c_str())){ a=atof(h.c_str());
73.         cout<<"NUMBER("<<i<<","<<t-h2+1<<") - "<<a<<endl;
74.     }
75.     else if(e1!=1){
76.         cout<<"IDENTIFIKATOR("<<i<<","<<t-h2+1<<") - "<<h<<endl;
77.         cout<<"\t\t\t\t DELIMETR("<<i<<","<<t+1<<") - "<<d<<endl;
78.     }
79.     h2=0; h.clear();
80.     }
81.     else if(d==" "){ s2=1; s=0; s1=0;
82.         float a;
83.         // raqam ekanligini aniqlash
84.         if(atof(h.c_str())){ a=atof(h.c_str());
85.             cout<<"NUMBER("<<i<<","<<t-h2+1<<") - "<<a<<endl;
86.         }
87.         else
88.             if(!h.empty()) cout<<"IDENTIFIKATOR("<<i<<","<<t-h2+1<<") -
"<<h<<endl;
89.             h2=0; h.clear();
90.         }
91.         else {
92.             h=h+d; h2++;}
93.         } // for siklini tugashi - bitta satrni tugashi
94.         // vazifadagi belgilar uchramagan satrlar
95.         if(s==0 && s1==0 && s2==0) {
96.             cout<<"IDENTIFIKATOR("<<i<<","<<i<<") - "<<line<<endl;
97.         }
98.         i++; // satrlar soni
99.     }
100.    getchar();
101.    return 0;
102.    }

```

Dastur natijasi:

```

C:\Users\User\Documents\Visual Studio 2012\Projects\Leksik_sintaksis_tahlil\Debug\Le
IDENTIFIKATOR(1,1) - #include<iostream>
IDENTIFIKATOR(2,2) - #include<fstream>
IDENTIFIKATOR(3,1) - using
IDENTIFIKATOR(3,7) - namespace
IDENTIFIKATOR(3,17) - std
                                DELIMETR(3,20) - ;
IDENTIFIKATOR(4,1) - int
IDENTIFIKATOR(4,5) - main(){
IDENTIFIKATOR(5,2) - float
IDENTIFIKATOR(5,8) - a
                                OPERATOR(5,9) - =
NUMBER(5,10) - 12.5
IDENTIFIKATOR(5,15) - b
                                OPERATOR(5,16) - =
NUMBER(5,17) - 4
IDENTIFIKATOR(6,4) - int
IDENTIFIKATOR(6,8) - c
                                OPERATOR(6,9) - =
IDENTIFIKATOR(6,10) - 0
                                DELIMETR(6,11) - ;
IDENTIFIKATOR(6,13) - c
                                OPERATOR(6,14) - +
IDENTIFIKATOR(6,15) -
                                OPERATOR(6,15) - =
IDENTIFIKATOR(6,16) - a
                                OPERATOR(6,17) - +

Arifmetik ifodada + ishora ketma-ket kelgan
IDENTIFIKATOR(6,19) - b
                                DELIMETR(6,20) - ;
IDENTIFIKATOR(6,22) - b
                                DELIMETR(6,23) - ;
IDENTIFIKATOR(7,4) - cout
IDENTIFIKATOR(7,9) - <<
IDENTIFIKATOR(7,12) - c
IDENTIFIKATOR(7,14) - <<
IDENTIFIKATOR(7,17) - endl
                                DELIMETR(7,21) - ;
IDENTIFIKATOR(8,8) - }

```

Matnli fayl (1.cpp) ma'lumotlari:

```

1  #include<iostream>
2  #include<fstream>
3  using namespace std;
4  int main(){
5      float a=12.5,b=4;
6      int c=0; c+=a++b; b;
7      cout << c << endl;
8  }

```

Matindagi xatolik (6- satr, 18- belgi) - dasturning natijasida “Arifmetik ifodada + ishora ketma-ket kelgan” xabari bilan izohlangan. Ushbu ishoradan

oldin ham + ishorasini **tokenlar jadvali – (OPERATOR(6,17) - +)** ko‘rinishida keltirilgan.

Dastur tahlili:

Kiruvchi ma’lumotlar matnli fayldan o‘zlashtirib olindi (**1.cpp**). Vazifada takidlangan belgilar (nuqta-vergul, **identifikatorlar, haqiqiy sonlar, o‘zlashtirish belgisi va figurali qavslar**) ajratib olindi. Qolgan belgilar esa IDENTIFIKATOR ko‘rinishida qabul qilindi. **atof()** funksiya bilan raqamga tekshirildi. Arifmetik ishoralar ketma-ketligi sintaksis tahlil qilindi – ya’niy har - xil xatolik bo‘ladigan vaziyatlar aniqlab olindi.

Masalan. *Ishora o‘rnida + ishorasi tanlandi.*

- 1) `c=a++b;`
- 2) `c=+a+b;`
- 3) `c=a+b++;`
- 4) `c=+++a+b;`
- 5) `c=a+b+;`
- 6) `a+;`
- 7) `+a;`
- 8) `+;`
- 9) `;;+;`
- 10) `,+;`
- 11) `+,`
- 12) `int a++;`

dastur 12 ta holat uchun sintaksis tahlil qiladi va ushbu 12 ta holat matnli faylda uchrasa xatolik haqida xabar chiqaradi.

Quyida Visual Studio ning Windows application muhitida tuzilgan tashqi ko‘rinishga ega bo‘lgan **leksik tahlil qiluvchi** dastur keltirilgan:

Dastur ko‘rinishi quyidagicha:

Lekoli va sintaktik tahlil dasturi

Ichma_ich.cpp

Identifikatorlar	Maxsus operatorlar	Ikali so'zlar	Operatorlar	Raqamlar
ostream	35	include		59
using	60	int		59
namespace	62	main	+	59
std	59	int	+	59
for	40	int		124
if	41	if		124
else	123	else	+	124
if	40	if	+	32
for	59	else		45
if	60	if		41
else	59	else		45
if	41			41
else	123			32
if	40			45
else	59			41
if	60			32
else	59			0

Identifikatorlar= 34
 Maxsus protsessorlar= 73
 Aktiv so'zlar= 11
 Operator bo'lmaganlar= 27
 Raqamlar= 16

Hisoblash

Daturning to'liq matni:

- 1) #pragma once
- 2) #include "stdafx.h"
- 3) #include <stdio.h>
- 4) #include <string.h>
- 5) #include <ctype.h>
- 6) #include <conio.h>
- 7) #define INT 01
- 8) #define CHAR 02
- 9) #define FLOAT 03
- 10) #define LONG 04
- 11) #define DOUBLE 05
- 12) #define STRUCT 06
- 13) #define PRINTF 07
- 14) #define SCANF 08
- 15) #define FOR 09
- 16) #define DO 10
- 17) #define WHILE 11
- 18) #define IF 12
- 19) #define ELSE 13
- 20) #define CONTINUE 14
- 21) #define BREAK 15
- 22) #define DEFAULT 16
- 23) #define CASE 17
- 24) #define GOTO 18
- 25) #define SWITCH 19

```

26)      #define OB          20
27)      #define CB          21
28)      #define LP          22
29)      #define RP          23
30)      #define EQUALTO     24
31)      #define LT          25
32)      #define GT          26
33)      #define PLUS        27
34)      #define MINUS       28
35)      #define MUL         29
36)      #define DIV         30
37)      #define COLON       31
38)      #define SCOLON      32
39)      #define TERNARY     33
40)      #define COMMA       34
41)      #define PLINE       35
42)      #define CARAT       36
43)      #define DIGIT       37
44)      #define ID          38
45)      #define OSB         39
46)      #define CSB         40
47)      #define HSH         41
48)      #define PER         42
49)      #define MAIN        43
50)      #define VOID        44
51)      #define NOTEQU      45
52)      #define AND         46
53)      #define BSH         47
54)      #define DQ          48
55)      #define SQ          49
56)      #define DOT         50
57)      #define INCLUDE     51
58)      #define SPACE       52
59)      #define TAB         53
60)      #define NUM_ENTRIES 54
61)      int token_no=0,key=0,ident=0,sp=0,op=0,digit=0;
62)      char
        keyM[100][100],identM[100][100],spM[10000],opM[100][100],digitM[1
        000];

63)      using namespace std;

64)      struct table
65)      {
66)      char    name[25];
67)      int     token;
68)      };

69)      struct table ref_tab[NUM_ENTRIES] = {

```

```

70) {"int", INT},
71) {"char", CHAR},
72) {"float", FLOAT},
73) {"long", LONG},
74) {"double", DOUBLE},
75) {"struct", STRUCT},
76) {"printf", PRINTF},
77) {"do", DO},
78) {"while", WHILE},
79) {"if", IF},
80) {"else", ELSE},
81) {"continue", CONTINUE},
82) {"break", BREAK},
83) {"default", DEFAULT},
84) {"case", CASE},
85) {"goto", GOTO},
86) {"switch", SWITCH},
87) {"{", OB},
88) {"}", CB},
89) {"(", LP},
90) {")", RP},
91) {"=", EQUALTO},
92) {"<", LT},
93) {">", GT},
94) {"+", PLUS},
95) {"-", MINUS},
96) {"*", MUL},
97) {"/", DIV},
98) {":", COLON},
99) {";", SCOLON},
100) {"?", TERNARY},
101) {"", COMMA},
102) {"|", PLINE},
103) {"^", CARAT},
104) {"[", OSB},
105) {"]", CSB},
106) {"#", HSH},
107) {"%", PER},
108) {"main", MAIN},
109) {"void", VOID},
110) {"!=", NOTEQU},
111) {"&", AND},
112) {"\\", BSH},
113) {"\"", DQ},
114) {"'", SQ},
115) {"include", INCLUDE},
116) };

```



```
117) char out_token[NUM_ENTRIES+1][40]=
118) {
119) " ",
120) "INT",
121) "CHAR",
122) "FLOAT",
123) "LONG",
124) "DOUBLE",
125) "STRUCT",
126) "PRINTF",
127) "SCANF",
128) "FOR",
129) "DO",
130) "WHILE",
131) "IF",
132) "ELSE",
133) "CONTINUE",
134) "BREAK",
135) "DEFAULT",
136) "CASE",
137) "GOTO",
138) "SWITCH",
139) "{",
140) "}",
141) "(",
142) ")",
143) "=",
144) "<",
145) ">",
146) "+",
147) "-",
148) "*",
149) "/",
150) ":",
151) ";",
152) "?",
153) ",",
154) "|",
155) "^",
156) "DIGIT",
157) "ID",
158) "[",
159) "]",
160) "#",
161) "%",
162) "MAIN",
163) "VOID",
164) "!=",
165) "&",
```

```

166)     "\\ ",
167)     "\"",
168)     "€",
169)     "•",
170)     "INCLUDE"
171) };
172) namespace Leksik_analizator {
173)     using namespace System;
174)     using namespace System::ComponentModel;
175)     using namespace System::Collections;
176)     using namespace System::Windows::Forms;
177)     using namespace System::Data;
178)     using namespace System::Drawing;
179)     /// <summary>
180)     /// Summary for Form1
181)     /// </summary>
182)     public ref class Form1 : public System::Windows::Forms::Form
183)     {
184)     public:
185)     Form1(void)
186)     {
187)     InitializeComponent();
188)     //
189)     //TODO: Add the constructor code here
190)     //
191)     }
192)     protected:
193)     /// <summary>
194)     /// Clean up any resources being used.
195)     /// </summary>
196)     ~Form1()
197)     {
198)     if (components)
199)     {
200)     delete components;
201)     }
202)     }
203)     private: System::Windows::Forms::DataGridView^
        dataGridView1;
204)     protected:
205)     private: System::Windows::Forms::DataGridViewTextBoxColumn^
        Column1;
206)     private: System::Windows::Forms::DataGridViewTextBoxColumn^
        Column2;
207)     private: System::Windows::Forms::DataGridViewTextBoxColumn^
        Column3;
208)     private: System::Windows::Forms::DataGridViewTextBoxColumn^
        Column4;

```

```

209)     private: System::Windows::Forms::DataGridViewTextBoxColumn^
        Column5;
210)     private: System::Windows::Forms::Button^    button1;
211)     private: System::Windows::Forms::TextBox^    textBox1;
212)     private: System::Windows::Forms::Label^    label1;
213)     private: System::Windows::Forms::Label^    label2;
214)     private: System::Windows::Forms::Label^    label3;
215)     private: System::Windows::Forms::Label^    label4;
216)     private: System::Windows::Forms::Label^    label5;
217)     private:
218)         /// <summary>
219)         /// Required designer variable.
220)         /// </summary>
221)     System::ComponentModel::Container ^components;

222)     #pragma region Windows Form Designer generated code
223)         /// <summary>
224)         /// Required method for Designer support - do not modify
225)         /// the contents of this method with the code editor.
226)         /// </summary>
227)     void InitializeComponent(void)
228)     {
229)         this->dataGridView1 = (gcnew
            System::Windows::Forms::DataGridView());
230)         this->Column1 = (gcnew
            System::Windows::Forms::DataGridViewTextBoxColumn());
231)         this->Column2 = (gcnew
            System::Windows::Forms::DataGridViewTextBoxColumn());
232)         this->Column3 = (gcnew
            System::Windows::Forms::DataGridViewTextBoxColumn());
233)         this->Column4 = (gcnew
            System::Windows::Forms::DataGridViewTextBoxColumn());
234)         this->Column5 = (gcnew
            System::Windows::Forms::DataGridViewTextBoxColumn());
235)         this->button1 = (gcnew System::Windows::Forms::Button());
236)         this->textBox1 = (gcnew System::Windows::Forms::TextBox());
237)         this->label1 = (gcnew System::Windows::Forms::Label());
238)         this->label2 = (gcnew System::Windows::Forms::Label());
239)         this->label3 = (gcnew System::Windows::Forms::Label());
240)         this->label4 = (gcnew System::Windows::Forms::Label());
241)         this->label5 = (gcnew System::Windows::Forms::Label());
242)         (cli::safe_cast<System::ComponentModel::ISupportInitialize^
            >(this->dataGridView1))->BeginInit();
243)         this->SuspendLayout();
244)         //
245)         // dataGridView1
246)         //

```

```

247)    this->dataGridView1->ColumnHeadersHeightSizeMode =
        System::Windows::Forms::DataGridViewColumnHeadersHeightSizeMode::
        AutoSize;
248)    this->dataGridView1->Columns->AddRange(gcnew cli::array<
        System::Windows::Forms::DataGridViewColumn^ >(5) {this->Column1,
249)    this->Column2, this->Column3, this->Column4, this-
        >Column5});
250)    this->dataGridView1->Location = System::Drawing::Point(12,
        54);
251)    this->dataGridView1->Name = L"dataGridView1";
252)    this->dataGridView1->Size = System::Drawing::Size(543, 422);
253)    this->dataGridView1->TabIndex = 0;
254)    //
255)    // Column1
256)    //
257)    this->Column1->HeaderText = L"Identifikatorlar";
258)    this->Column1->Name = L"Column1";
259)    //
260)    // Column2
261)    //
262)    this->Column2->HeaderText = L"Maxsus operatorlar";
263)    this->Column2->Name = L"Column2";
264)    //
265)    // Column3
266)    //
267)    this->Column3->HeaderText = L"kalit sozlar";
268)    this->Column3->Name = L"Column3";
269)    //
270)    // Column4
271)    //
272)    this->Column4->HeaderText = L"Operatorlar";
273)    this->Column4->Name = L"Column4";
274)    //
275)    // Column5
276)    //
277)    this->Column5->HeaderText = L"Raqamlar";
278)    this->Column5->Name = L"Column5";
279)    //
280)    // button1
281)    //
282)    this->button1->Location = System::Drawing::Point(153, 482);
283)    this->button1->Name = L"button1";
284)    this->button1->Size = System::Drawing::Size(222, 35);
285)    this->button1->TabIndex = 1;
286)    this->button1->Text = L"Hisoblash";
287)    this->button1->UseVisualStyleBackColor = true;
288)    this->button1->Click += gcnew System::EventHandler(this,
        &Form1::button1_Click);
289)    //

```

```

290) // textBox1
291) //
292) this->textBox1->Font = (gcnew
    System::Drawing::Font(L"Microsoft Sans Serif", 18.25F,
    System::Drawing::FontStyle::Regular,
    System::Drawing::GraphicsUnit::Point,
293) static_cast<System::Byte>(204));
294) this->textBox1->ForeColor =
    System::Drawing::SystemColors::MenuText;
295) this->textBox1->Location = System::Drawing::Point(12, 12);
296) this->textBox1->Multiline = true;
297) this->textBox1->Name = L"textBox1";
298) this->textBox1->Size = System::Drawing::Size(543, 36);
299) this->textBox1->TabIndex = 2;
300) this->textBox1->Text = L"Ichma_ich.cpp";
301) this->textBox1->Click += gcnew System::EventHandler(this,
    &Form1::textBox1_Click);
302) this->textBox1->TextChanged += gcnew
    System::EventHandler(this, &Form1::textBox1_TextChanged);
303) //
304) // label1
305) //
306) this->label1->AutoSize = true;
307) this->label1->Font = (gcnew
    System::Drawing::Font(L"Microsoft Sans Serif", 18.25F,
    System::Drawing::FontStyle::Regular,
    System::Drawing::GraphicsUnit::Point,
308) static_cast<System::Byte>(204));
309) this->label1->Location = System::Drawing::Point(607, 41);
310) this->label1->Name = L"label1";
311) this->label1->Size = System::Drawing::Size(81, 29);
312) this->label1->TabIndex = 3;
313) this->label1->Text = L"label1";
314) //
315) // label2
316) //
317) this->label2->AutoSize = true;
318) this->label2->Font = (gcnew
    System::Drawing::Font(L"Microsoft Sans Serif", 18.25F,
    System::Drawing::FontStyle::Regular,
    System::Drawing::GraphicsUnit::Point,
319) static_cast<System::Byte>(204));
320) this->label2->Location = System::Drawing::Point(607, 92);
321) this->label2->Name = L"label2";
322) this->label2->Size = System::Drawing::Size(81, 29);
323) this->label2->TabIndex = 3;
324) this->label2->Text = L"label2";
325) //
326) // label3

```

```

327) //
328) this->label3->AutoSize = true;
329) this->label3->Font = (gcnew
    System::Drawing::Font(L"Microsoft Sans Serif", 18.25F,
    System::Drawing::FontStyle::Regular,
    System::Drawing::GraphicsUnit::Point,
330) static_cast<System::Byte>(204));
331) this->label3->Location = System::Drawing::Point(607, 143);
332) this->label3->Name = L"label3";
333) this->label3->Size = System::Drawing::Size(81, 29);
334) this->label3->TabIndex = 3;
335) this->label3->Text = L"label3";
336) //
337) // label4
338) //
339) this->label4->AutoSize = true;
340) this->label4->Font = (gcnew
    System::Drawing::Font(L"Microsoft Sans Serif", 18.25F,
    System::Drawing::FontStyle::Regular,
    System::Drawing::GraphicsUnit::Point,
341) static_cast<System::Byte>(204));
342) this->label4->Location = System::Drawing::Point(607, 197);
343) this->label4->Name = L"label4";
344) this->label4->Size = System::Drawing::Size(81, 29);
345) this->label4->TabIndex = 3;
346) this->label4->Text = L"label4";
347) //
348) // label5
349) //
350) this->label5->AutoSize = true;
351) this->label5->Font = (gcnew
    System::Drawing::Font(L"Microsoft Sans Serif", 18.25F,
    System::Drawing::FontStyle::Regular,
    System::Drawing::GraphicsUnit::Point,
352) static_cast<System::Byte>(204));
353) this->label5->Location = System::Drawing::Point(607, 254);
354) this->label5->Name = L"label5";
355) this->label5->Size = System::Drawing::Size(81, 29);
356) this->label5->TabIndex = 3;
357) this->label5->Text = L"label5";
358) //
359) // Form1
360) //
361) this->AutoScaleDimensions = System::Drawing::SizeF(6, 13);
362) this->AutoScaleMode =
    System::Windows::Forms::AutoScaleMode::Font;
363) this->ClientSize = System::Drawing::Size(1027, 529);
364) this->Controls->Add(this->label5);
365) this->Controls->Add(this->label4);

```

```

366) this->Controls->Add(this->label3);
367) this->Controls->Add(this->label2);
368) this->Controls->Add(this->label1);
369) this->Controls->Add(this->textBox1);
370) this->Controls->Add(this->button1);
371) this->Controls->Add(this->dataGridView1);
372) this->Name = L"Form1";
373) this->Text = L"Leksik va sintaktik tahlil dasturi";
374) (cli::safe_cast<System::ComponentModel::ISupportInitialize^
    >(this->dataGridView1))->EndInit();
375) this->ResumeLayout(false);
376) this->PerformLayout();

377) }
378) #pragma endregion

379) int search(char lexeme[])
380) {
381) int i;

382) for ( i = 0; i < NUM_ENTRIES; i++)
383) {
384) if (strcmp(lexeme,ref_tab[i].name) == 0)
385) {
386) strcpy_s(keyM[key],lexeme);
387) key++;
388) return ref_tab[i].token;
389) }
390) /* else
391) {
392) ident++;
393) }
394) */

395) }
396) strcpy_s(identM[ident],lexeme);
397) ident++;
398) return ID;
399) }

400) void output(int num)
401) {
402) //printf("\n%d\t%s\t\n",++token_no,out_token[num]);
403) /* delay(350); */
404) }

405) void lexical(char store[], int store_len)
406) {
407) int i,j,line=2;

```

```

408) char ch,lexeme[10],next;
409) for (i = 0; i < store_len;)
410) {
411) ch = store[i];
412) switch(ch)
413) {
414) case ' ':
415) i++;
416) printf("\tSPACE REMOVED\n");
417) break;
418) case '{':
419) i++;
420) spM[sp]='{';
421) sp++;
422) //output(OB);
423) break;
424) case '}':
425) i++;
426) spM[sp]='}';
427) sp++;
428) //output(CB);
429) break;
430) case '(':
431) i++;
432) spM[sp]='(';
433) sp++;
434) //output(LP);
435) break;
436) case ')':
437) i++;
438) spM[sp]=')';
439) sp++;
440) //output(RP);
441) break;
442) case '=':
443) i++;
444) spM[sp]='=';
445) op++;
446) //output(EQUALTO);
447) break;
448) case '<':
449) i++;
450) spM[sp]='<';
451) sp++;
452) //output(LT);
453) break;
454) case '>':
455) i++;
456) spM[sp]='>'; sp++;
457) //output(GT);
458) break;
459) case '!':
460) i++;
461) spM[sp]='!';
462) sp++;
463) //output(NOTEQU);
464) break;
465) case '+':
466) i++;
467) opM[op][0]='+';
468) op++;
469) //output(PLUS);
470) break;
471) case '-':
472) i++;
473) opM[op][0]='-';
474) op++;
475) //output(MINUS);
476) break;
477) case '*':
478) i++;
479) opM[op][0]='*';
480) op++;
481) //output(MUL);
482) break;
483) case '/':
484) i++;
485) opM[op][0]='/';
486) op++;
487) next=store[i];
488) if(next=='*')

```



```

489)     {
490)     i++;
491)     while(store[i]!='*')i++;
492)     i++;
493)     if(store[i]=='/')
494)     // else goto
495)     printf("\n\tComments
removed\n");
496)     i++;
497)     }
498)     else if(next=='/')
499)     {
500)     i++;
501)     while(store[i]!='\n')i
++;
502)     i++;
503)     printf("\n\tComments
removed\n");
504)     }
505)     else{
506)     opM[op][0]='\n';
507)     op++;
508)     }
509)     //output(DIV);
510)     break;
511)     case ':':
512)     i++;
513)     spM[sp]=': ';
514)     sp++;
515)     //output(COLON);
516)     break;
517)     case ';':
518)     i++;
519)     spM[sp]='; ';
520)     sp++;
521)     //output(SCOLON);
522)     break;
523)     case '?':
524)     i++;
525)     spM[sp]='+ ';
526)     sp++;
527)     //output(TERNARY);
528)     break;
529)     case '\n':
530)     i++;
531)     spM[sp]=ch;
532)     sp++;
533)     //output(DQ);
534)     break;
535)     case '\\':
536)     i++;
537)     spM[sp]=ch;
538)     sp++;
539)     //output(SQ);
540)     break;
541)     case ',':
542)     i++;
543)     spM[sp]=ch;
544)     sp++;
545)     //output(COMMA);
546)     break;
547)     case '|':
548)     i++;
549)     spM[sp]=ch;
550)     sp++;
551)     //output(PLINE);
552)     break;
553)     case '^':
554)     i++;
555)     spM[sp]=ch;
556)     sp++;
557)     //output(CARAT);
558)     break;
559)     case '[':
560)     i++;
561)     spM[sp]=ch;
562)     sp++;
563)     output(OSB);
564)     break;
565)     case ']':
566)     i++;
567)     spM[sp]=ch;
568)     sp++;
569)     output(CSB);
570)     break;
571)     case '#':
572)     i++;
573)     spM[sp]=ch;
574)     sp++;
575)     output(HSH);
576)     break;
577)     case '%':
578)     i++;
579)     spM[sp]=ch;
580)     sp++;
581)     output(PER);
582)     break;
583)     /* case '!':

```

```

584)     i++;
585)     sp++;
586)     output(NOTEQU);
587)     break; */
588)     case '&':
589)     i++;
590)     spM[sp]=ch;
591)     sp++;
592)     output(AND);
593)     break;
594)     case '\\':
595)     i++;
596)     spM[sp]=ch;
597)     sp++;
598)     output(BSH);
599)     break;
613)     output(search(lexeme));
614)     break;
615)     }
616)     else if(isdigit(store[i]))
617)     {
618)     j = 0;
619)     while (isdigit(store[i]))
620)     lexeme[j++] = store[i++];
621)     lexeme[j] = '\0';
622)     digitM[digit]=store[i];
623)     digit++;
624)     output(DIGIT);
625)     break;
626)     }
627)     else if (store[i]=='\n')
628)     {
629)     i++;
630)     printf("\n Line=%d\n",line++);
631)     printf("  ____\n");
632)     }
633)     else if (store[i]=='\t' || store[i]==' ')
634)     i++;
635)     else
636)     i++, printf("\tInvalid symbol\n");
637)     } } }
638)     private: System::Void button1_Click(System::Object^ sender,
        System::EventArgs^ e) {
639)     FILE *fp1; /* Source file pointer */
640)     int ch,i; /* used to store character read */
641)     char store[4000];
642)     int store_len;
643)     char file[15];
644)     /* clrscr(); */
600)     case '.':
601)     i++;
602)     spM[sp]=ch;
603)     sp++;
604)     output(DOT);
605)     break;
606)     default:
607)     if (isalpha(store[i]))
608)     {
609)     j = 0;
610)     while(isalpha(store[i]
        ))
611)     lexeme[j++] =
        store[i++];
612)     lexeme[j] = '\0';

```

```

645) printf("Enter file name with .c as the extension\n");
646) scanf("%s",file);
647) //strcat(file, textBox1->Text->);
648) for(int i=0; i<textBox1->Text->Length; i++)// i ni textBox1
    ni uzunligigacha 1 tadan oshirish
649) { file[i]=textBox1->Text[i]; }
650) fp1 = fopen(file, "r");
651) if (fp1 == NULL)
652) {
653) //printf("The source file can not be opened for reading\n");
654) MessageBox::Show("Fayl topilmadi");
655) /*exit(1); */
656) }
657) else {
658) /* read till end of file is encountered */
659) i = 0;
660) while ( (ch = getc(fp1)) != EOF)
661) {
662) store[i++] = ch;
663) }
664) store_len = i;
665) /* close the input file */
666) fclose(fp1);
667) lexical(store, store_len);
668) label1->Text = "Identifikatorlar= "+ident.ToString();
669) dataGridView1->RowCount=ident;
670) for (int i = 0; i < ident; i++)
671) {
672) String ^s = gcnew String(identM[i]);
673) //dataGridView1->Rows->Add();
674) dataGridView1->Rows[i]->Cells[0]->Value =s;
675) }
676) //printf("identifiers=%d",ident);
677) if( dataGridView1->RowCount<sp){dataGridView1->RowCount=sp;}
678) label2->Text = "Maxsus protsessorlar= "+sp.ToString();
679) for (int i = 0; i < sp; i++)
680) {
681) String ^s = Convert::ToString(spM[i]);
682) //dataGridView1->Rows->Add();
683) dataGridView1->Rows[i]->Cells[1]->Value =s;
684) }
685) //printf("\n\n special operators=%d",sp);
686) label3->Text = "Aktivr so'zlar= "+key.ToString();
687) if( dataGridView1->RowCount<key){dataGridView1-
    >RowCount=key;}
688) for (int i = 0; i < key; i++)
689) {
690) String ^s = gcnew String(keyM[i]);

```

```

691) //dataGridView1->Rows->Add();
692) dataGridView1->Rows[i]->Cells[2]->Value =s;
693) }
694) //printf("\n\nkeywords=%d",key);
695) if( dataGridView1->RowCount<op){dataGridView1->RowCount=op;}
696) for (int i = 0; i < op; i++)
697) {
698) String ^s = gcnew String(opM[i]);
699) //dataGridView1->Rows->Add();
700) dataGridView1->Rows[i]->Cells[3]->Value =s;
701) }
702) label4->Text = "Operator bo‘lmaganlar= "+op.ToString();
703) //printf("\n\noperators=%d",op);
704) if( dataGridView1->RowCount<digit){dataGridView1-
>RowCount=digit;}
705) for (int i = 0; i < op; i++)
706) {
707) String ^s = Convert::ToString(digitM[i]);
708) //dataGridView1->Rows->Add();
709) dataGridView1->Rows[i]->Cells[4]->Value =s;
710) }
711) label5->Text = "Raqamlar= "+digit.ToString();
712) //printf("\n\ndigit=%d\n",digit);
713) }
714) //getch();
715) }
716) private: System::Void textBox1_TextChanged(System::Object^
sender, System::EventArgs^ e) {
717) }
718) private: System::Void textBox1_Click(System::Object^
sender, System::EventArgs^ e) {
719) textBox1->Clear();
720) } };
721) }

```

Laboratoriya mashg‘ulotini bajarish bo‘yicha variantlar

Quyida berilgan vazifalarda kiruvchi qiymatlar matnli fayldan o‘zlashtirilishi kerak.

1) Topshiriqqa muvofiq kirish matnini **leksik** tahlilini bajaradigan va ularning turlarini ko‘rsatgan holda **tokenlar jadvalini** tuzadigan dastur yozing. Dastur kirish matnida **leksik tahlil** bosqichida aniqlanishi mumkin bo‘lgan xatolar (**sintaksis tahlil**) mavjudligi to‘g‘risida xabarlarni chiqarishi kerak;

2) **leksik va sintaksis tahlil** analizator kodini yaratishda Visual C++ dan kompilyator sifatida foydalaning.

7- jadval

1	Matinli faylda “ ; ” (nuqta-vergul) belgisi bilan ajratilgan mantiqiy iboralar mavjud. Mantiqiy iboralar: identifikatorlardan, 0 va 1 konstatalaridan, o‘zlashtirish belgisi (=), operatsiyalar or, xor, and, nor, , &&, !A va figurali qavs({}) lardan iborat.
2	Matinli faylda “ ; ” (nuqta-vergul) belgisi bilan ajratilgan shartli operatorlar (if(), else if(), if().else) mavjud. Shartli operatorlar: identifikatorlardan, solishtirish belgilari (<, >, =, <=, >=, !=) dan, o‘nlik suzuvchi nuqta raqamlaridan, o‘zlashtirish belgisi (=), operatsiyalar or, xor, and, nor, , &&, !A va figurali qavs({}) lardan iborat.
3	Matinli faylda “ ; ” (nuqta-vergul) belgisi bilan ajratilgan sikl operatorlar (for(..., ..., ...){}) mavjud. Sikl operatorlar: identifikatorlardan, solishtirish belgilari (<, >, =, <=, >=, !=) dan, satr konstantalari (ikki qatorli belgilar ketma-ketligi) raqamlaridan, o‘zlashtirish belgisi (=), va figurali qavs({}) lardan iborat.
4	Matinli faylda “ ; ” (nuqta-vergul) belgisi bilan ajratilgan sikl operatorlar (while(...){}) mavjud. Sikl operatorlar: identifikatorlardan, solishtirish belgilari (<, >, =, <=, >=, !=) dan, satr konstantalari (ikki qatorli belgilar ketma-ketligi) raqamlaridan, o‘zlashtirish belgisi (=), va figurali qavs({}) lardan iborat.
5	Matinli faylda “ ; ” (nuqta-vergul) belgisi bilan ajratilgan sikl operatorlar (do...while(...){}) mavjud. Sikl operatorlar: identifikatorlardan, solishtirish belgilari (<, >, =, <=, >=, !=) dan, X, V katta harflar bilan yozilgan Rim raqamlaridan, o‘zlashtirish belgisi (=), va figurali qavs({}) lardan iborat.
6	Matinli faylda “ ; ” (nuqta-vergul) belgisi bilan ajratilgan arifmetik iboralar mavjud. Arifmetik ifodalar: identifikatorlardan, o‘nlik suzuvchi nuqta raqamlaridan, o‘zlashtirish belgisi (=), X, V katta harflar bilan yozilgan Rim raqamlaridan,), ishoralar +, -, *, / va figurali qavs({}) lardan iborat.
7	Matinli faylda “ ; ” (nuqta-vergul) belgisi bilan ajratilgan mantiqiy iboralar mavjud. Mantiqiy iboralar: identifikatorlardan, false va true

	konstataridan, o‘zlashtirish belgisi (=), operatsiyalar or, xor, and, nor, , &&, !A va figurali qavs({}) lardan iborat.
8	Matinli faylda “ ; ” (nuqta-vergul) belgisi bilan ajratilgan massiv elementlari mavjud. Massiv elementlari: identifikatorlardan, o‘nlik suzuvchi nuqta raqamlaridan, o‘zlashtirish belgisi (=), ishoralar +, -, *, / va figurali qavs({}) lardan iborat.
9	Matinli faylda “ ; ” (nuqta-vergul) belgisi bilan ajratilgan ko‘p o‘lchovli massiv elementlari mavjud. Ko‘p o‘lchovli massiv elementlari: identifikatorlardan, o‘nlik suzuvchi nuqta raqamlaridan, o‘zlashtirish belgisi (=), ishoralar +, -, *, / va figurali qavs({}) lardan iborat.
10	Matinli faylda “ ; ” (nuqta-vergul) belgisi bilan ajratilgan funksiya prototiplari mavjud. Funksiya prototiplari: local va global identifikatorlardan, o‘nlik suzuvchi nuqta raqamlaridan, o‘zlashtirish belgisi (=), ishoralar +, -, *, / va figurali qavs({}) lardan iborat.
11	Matinli faylda “ ; ” (nuqta-vergul) belgisi bilan ajratilgan rekursiv funksiyalar mavjud. Rekursiv funksiyalar: local va global identifikatorlardan, o‘nlik suzuvchi nuqta raqamlaridan, o‘zlashtirish belgisi (=), qiymat qaytarish aktiv so‘zi return, ishoralar +, -, *, / va figurali qavs({}) lardan iborat.
12	Matinli faylda “ ; ” (nuqta-vergul) belgisi bilan ajratilgan ko‘rsatgichlar mavjud. Ko‘rsatgichlar: identifikatorlardan, o‘nlik suzuvchi nuqta raqamlaridan, o‘zlashtirish belgisi (=), adres olish belgi (&), ishoralar +, -, *, / va figurali qavs({}) lardan iborat.
13	Matinli faylda “ ; ” (nuqta-vergul) belgisi bilan ajratilgan struktura mavjud. Struktura: identifikatorlardan, o‘nlik suzuvchi nuqta raqamlaridan, o‘zlashtirish belgisi (=), adres olish belgisi (&), ishoralar +, -, *, / va figurali qavs({}) lardan iborat.
14	Matinli faylda “ ; ” (nuqta-vergul) belgisi bilan ajratilgan class mavjud. Class: konsturtorlardan, identifikatorlardan, o‘zlashtirish belgisi (=), adres olish belgisi (&), ishoralar +, -, *, / va figurali qavs({}) lardan iborat.
15	Matinli faylda “ ; ” (nuqta-vergul) belgisi bilan ajratilgan class mavjud. Class: class a‘zolari va class funksiya a‘zolaridan, o‘zlashtirish belgisi (=), adres olish belgisi (&), ishoralar +, -, *, / va figurali qavs({}) lardan iborat.
16	Matinli faylda “ ; ” (nuqta-vergul) belgisi bilan ajratilgan class mavjud. Class: do‘st (friend) funksiyalardan, o‘zlashtirish belgisi (=), adres olish belgisi (&), ishoralar +, -, *, / va figurali qavs({}) lardan iborat.

17	Matinli faylda “ ; ” (nuqta-vergul) belgisi bilan ajratilgan funksiyalar mavjud. Funksiyalar: local va global identifikatorlardan, parametrlardan, o‘zlashtirish belgisi (=), adres olish belgisi (&), ishoralar +, -, *, / va figurali qavs({}) lardan iborat.
18	Matinli faylda “ ; ” (nuqta-vergul) belgisi bilan ajratilgan qayta yuklangan funksiyalar mavjud. Qayta yuklangan funksiyalar: local va global identifikatorlardan, xar-xil turdagi parametrlardan, o‘zlashtirish belgisi (=), adres olish belgisi (&), ishoralar +, -, *, / va figurali qavs({}) lardan iborat.
19	Matinli faylda “ ; ” (nuqta-vergul) belgisi bilan ajratilgan voris classlar mavjud. Voris class lar: identifikatorlar, void turidagi funksiyalar, o‘zlashtirish belgisi (=), adres olish belgisi (&), ishoralar +, -, *, / va figurali qavs({}) lardan iborat.
20	Matinli faylda “ ; ” (nuqta-vergul) belgisi bilan ajratilgan voris classlar mavjud. Voris class lar: identifikatorlar, qiymat qaytaruvchi funksiyalar, o‘zlashtirish belgisi (=), adres olish belgisi (&), ishoralar +, -, *, / va figurali qavs({}) lardan iborat.

Takrorlash uchun savol va topshiriqlar:

1. Leksik tahlil nima?
2. Sintaksis tahlil nima?
3. Semantik tahlil nima?
4. Token nima?
5. Kompilyator nima?
6. Kompilyatsiya nima?
7. Leksik tahli dasturning qanday bosqichida amalga oshiriladi?
8. Sintaksis tahli dasturning qanday bosqichida amalga oshiriladi?
9. Semantik tahli dasturning qanday bosqichida amalga oshiriladi?
10. Tokendan nima maqsadda foydalaniladi?
11. Leksema nima?
12. Leksemalarni asosiy ko‘rinishlari qanday ko‘rinishda bo‘ladi?

5- LABORATORIYA ISHI

5. MURAKKAB SARALASH ALGORITMLARI VA KATTA SONLAR BILAN ISHLASH

5.1. Katta sonlar bilan ishlovchi kutubxonasi va uning funksiyalariga doir amaliy dasturlar tuzish

Multiprecision kutubxonasi oddiy C ++ tipidagi turlarga qaraganda kattaroq diapazon va aniqlikka ega bo'lgan C ++ da butun, haqiqiy va suzuvchi nuqtalarni taqdim etadi. Multiprecision-dagi katta sonlarning turlaridan asosiy matematik operatsiyalar, elementar transsendental funktsiyalar, shuningdek Boost.Math-dagi vazifalar bilan foydalanish mumkin. Multiprecision turlari shuningdek, aniq tarjima qilingan qoidalar yordamida C ++-da o'rnatilgan turlar bilan o'zaro ta'sir qilishi mumkin. Bu sizga Boost.Multiprecision-dan har qanday matematik hisob-kitoblarni, shu jumladan butun sonlarni, ratsional turlarni va suzuvchi nuqta raqamlarini, keng doirani va aniqlikni talab qiladigan narsalarni ishlatishga imkon beradi.

Multiprecision ko'p sonli matematiklar uchun universal interfeysdan iborat, shuningdek butun, oqilona va suzuvchi turlarni qo'llab-quvvatlanadigan ko'p sonli server qismlarini tanlash uchun. Boost.Multiprecision oflayn rejimida taqdim etiladigan server qismlarini tanlashni, shu jumladan GMP, MPFR, MPIR, TomMath uchun interfeyslarni, shuningdek o'zlarining litsenziyalangan Boost to'plamini, butun sonlar, ratsional sonlar va suzuvchi nuqta raqamlari uchun faqat server qismi sarlavhalari uchun taqdim etadi. Bunga qo'shimcha ravishda, serverning shaxsiy qismlari Multiprecision interfeysi yordamida yaratilishi va ishlatilishi mumkin, bu holda sinf amalga oshirish zarur tushunchalarga mos keladi.

Raqam turiga qarab, aniqlik o'zboshimchalik bilan katta bo'lishi mumkin (faqat mavjud xotira bilan cheklangan), kompilyatsiya vaqtida o'rnatilishi mumkin (masalan, 50 yoki 100 ta o'nlik raqamlar) yoki ish vaqtida a'zo funktsiyalari tomonidan boshqariladigan o'zgaruvchi. Ifodali naqshli turlari oddiy odatiy turlarga qaraganda yaxshiroq ishlashni ta'minlaydi.

Multiprecision kutubxonasi ikki qismdan iborat:

Ekspressiya shablonlari uchun qo'llab-quvvatlanadigan tashqi interfeys, operatorning barcha ortiqcha ishlovlarini bajaradigan, ifoda baholashni optimallashtiradigan va kodni kamaytiradigan raqam.

Haqiqiy arifmetik operatsiyalarni bajaradigan va faqat tashqi interfeys interfeysining qisqartirilgan talablariga javob beradigan server qismlarini tanlash.

Tashqi interfeys va ichki interfeysni ajratish sizga iloji boricha yuqori darajada rivojlangan, ammo cheklangan litsenziyali kutubxonalardan foydalanishga imkon beradi, ammo ko'chma cheksiz litsenziyaga ega foydalanuvchilar uchun Boost alternativalarini taqdim etadi. Boshqacha qilib aytganda, ba'zi manbalar uchinchi tomon kutubxonalariga tayanadi, ammo Boost versiyasi har doim faqat sarlavhalar uchun mavjud (agar u biroz sekinroq bo'lsa).

Agar siz Boost litsenziyasi bilan to'liq sonli turni qidirish va undan foydalanishni istasangiz, bir nechta aniq sonlar uchun `cpp_int` ga, ko'p aniqlikdagi suzuvchi nuqta turlari uchun `cpp_dec_float`-ga va ratsional turlar uchun `cpp_rational`-ga o'ting.

Kutubxona ko'pincha oldindan belgilangan turlardan biri orqali ishlatiladi: masalan, agar siz bazani amalga oshirish uchun GMP-dan foydalanib o'zboshimchalik bilan aniqlik turini ishlatmoqchi bo'lsangiz, quyidagilarni ishlatishingiz mumkin:

```
#include < boost / multiprecision / gmp . hpp > // GMP kutubxonasi turlarini uchun
```

```
boost :: multiprecision :: mpz_int myint ; // Ixtiyoriy butun tur uchun
```

Bunga qo'shimcha ravishda, siz o'zingizning ko'p qirrali aniqlik turingizni yaratishingiz mumkin, bu raqamni oldindan belgilangan ichki turlardan biriga qo'shish orqali amalga oshiriladi. Masalan, siz MPFR kutubxonasiga asoslangan 300 ta o'nlik joy bilan suzuvchi nuqta turini xohlaysiz deylik. Bunday holda, ushbu aniqlik darajasi bilan oldindan belgilangan tipedef mavjud emas, shuning uchun biz o'zimiznikini yaratamiz:

```
#include < boost / multiprecision / mpfr . hpp > // MPFR-ni o'zgartiradigan  
belgi turini belgilaydi  
nom maydoni mp = boost :: multiprecision ; // keying mavzularda korib  
chiqiladi  
my_float a , b , c ; // Ushbu o'zgaruvchilar 300 ta o'nlik raqamgacha aniqlikka  
ega.
```

Yuqoridagi misolni takrorlashimiz mumkin, ammo ifoda shablonlari o'chirilgan holda (tezroq tuzish vaqti uchun, lekin sekinroq bajarish vaqti bilan), ikkinchi dalilni raqam shabloniga o'tkazib yuboramiz:

Arifmetik amallarni har xil turlarga aralashtirishimiz mumkin, agar bitta turdan ikkinchisiga aniq bir konversiya mavjud bo'lsa:

```
#include <boost/multiprecision/cpp_int.hpp>  
namespace mp = boost::multiprecision; //  
mp::int128_t a(3), b(4);  
mp::int512_t c(50), d;  
d = c * a; // aralash arifmetikaning natijasi int512_t
```

O'zgarishlarga ham ruxsat beriladi:

```
d = a;
```

```
d = a * b;
```

Ammo, tabiiyki, zararli bo'lgan konversiyalar aniq yoki to'liq taqiqlangan:

```
d = 3.14; - xatolik
```

```
d = static_cast<mp::int512_t>(3.14); - ruxsat beriladi
```

Agar konvertatsiya noaniq yoki aniq bo'lsa, aralash arifmetika bajarilmaydi:

```
number<cpp_int_backend<>, et_off> a(2);
```

```
number<cpp_int_backend<>, et_on> b(3);
```

```
b = a * b; // Xato, yashirin konversiya har ikki tomon ham ketishi mumkin.
```

```
b = a * 3.14; // Xato, agar konversiya aniq bo'lsa, hech qanday operator  
ortiqcha yuklamaydi.
```

Ishning maqsadi: Katta sonlar ustida arifmetik amallar bajarish.

Masalaning qo‘yilishi:

Katta sonlar ustida arifmetik amallarni bajarishda xotitira hajmining sarflanishini aniqlash.

Masala: 86 xonali 2 ta sonni ko‘paytiruvchi, bo‘luvchi va ayruvchi dastur tuzing.

Masalani yechish g‘oyasi: boost. Multiprecision kutubxonasidagi funksiyalaridan foydalanib, masala yechiladi.

```
#include <boost/multiprecision/cpp_dec_float.hpp>
```

```
#include <boost/multiprecision/cpp_int.hpp>
```

Quyida keltirilgan dasturda katta sonlar bilan ishlashchi funksiyalar joylashgan “**boost.multiprecision**” kutubxondan foydalanilgan. Ushbu kutubxona quyidagi tartibda tizimga o‘rnatiladi:

1. Visual Studio uchun boost ni sozlash

Visual Studio 12 uchun boostning 1.52 versiyasini yoki yangi versiyasini quyidagi havoladan yuklab olishingiz mumkin:

Ohirgi versiyasi: <http://www.boost.org/users/download>.

1.54 versiyasi: <http://sourceforge.net/projects/boost/files/boost/1.54.0/>

2. Kompilyatsiya:

Kutubxonaning “**1.54.zip**” nomli fayli internetdan yuklab olingandan keyin, C:\ diskka shu nom bilan arxivdan chiqariladi va C diskga boost nomli katalog yaratiladi. Boost katalogining ichiga 1.54 katalogi yaratildi. Uning ichiga arxivdan chiqqan katalog “**1.54**” тиртп ichidagi hamma fayllar va kataloglar nushalanadi-ko‘chiriladi. Misol: “**C:\boost\1.54**”.

Komanda stroka (Win+R) kadan quyidagi buyrug‘ beriladi:

```
C:\boost\1.54\bootstrap.bat
```

Buyrug‘ muvaffaqiyatli amalga oshgandan keyin, kompilyatsiya jarayonini boshlash mumkin. Komanda stroka (Win+R) kadan quyidagi buyrug‘ beriladi:

b2

Kompilyatsiya jarayoni 20 minut davot etadi. Hamda quyida xabar chiqariladi:

The Boost C++ Libraries were successfully built!

The following directory should be added to compiler include paths:

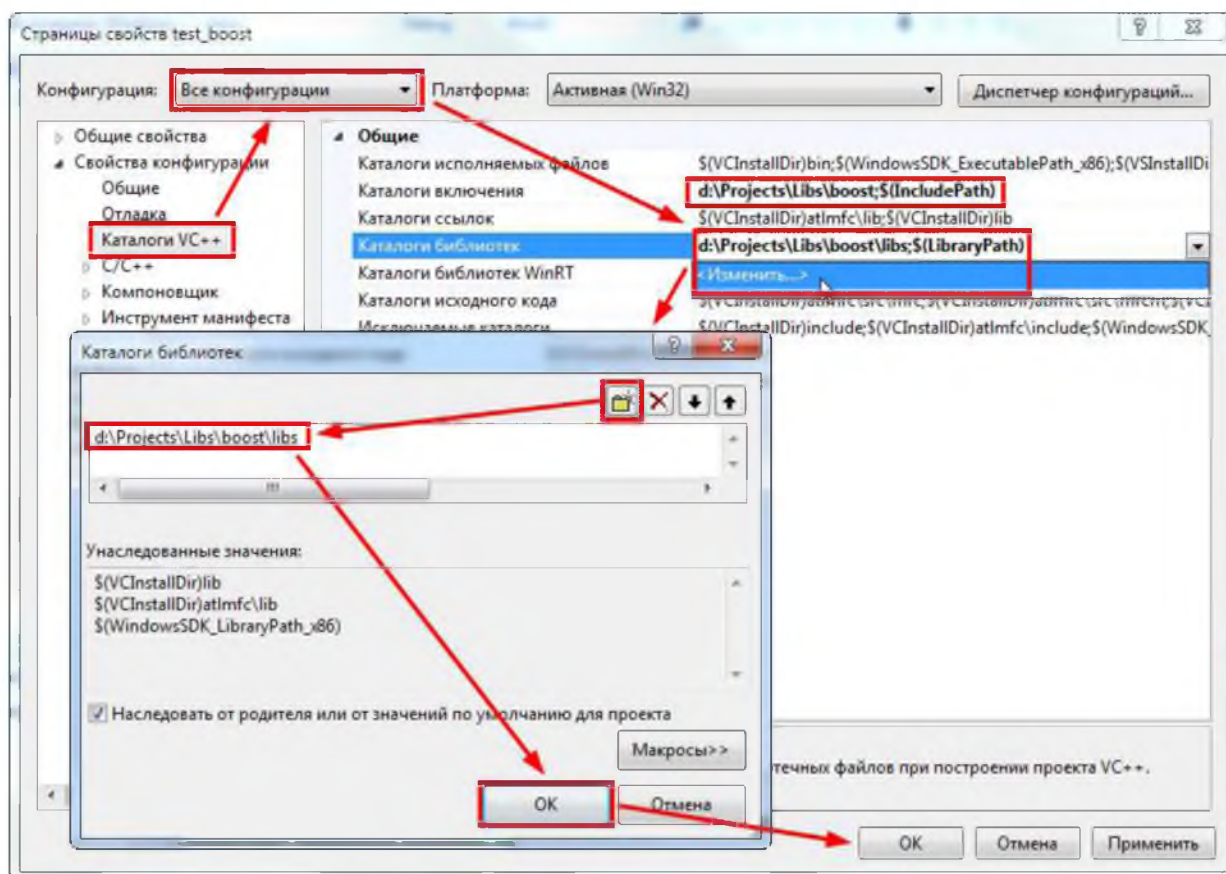
C:\boost\1.54

The following directory should be added to linker library paths:

C:\boost\1.54\stage\lib

3. Visual Studio 12 da projekt yaratish va uni sozlash

Buning uchun “win32 konsole ilovasi” dan foydalanamiz. Projekt yaratilgandan keyin, Visual Studio 12 ishchi muhitining chap taraftida projektning hususiyati (свойства) ga kiriladi hamda kutubxona kompilyatsiya qilingan kataloglar kompilyatsiya qilish uchun ko‘rsatib qo‘yiladi. Ushbu jarayon quyidagi rasmda keltirilgan:



3.1- rasm. Visaul Studioni sozlash jarayoni

F5 tugmasi bosilib, projekt kompilyatsiya qilinadi.

2432902008176640000

933262154439441526816992388562667004907159682643816214685929638952
175999932299156089414639761565182862536979208272237582511852109168
64000000000000000000000000000000

5.2. Murakkab saralash algoritmlari

Murakkab saralash algoritmlari haqida.

Vaqtning toʻrtidan bir qismiga qadar markazlashtirilgan kompyuterlarga saralash maʼlumotlari berilishi hisoblab chiqilgan. Buning sababi, oldindan saralangan massivda qiymatni topish osonroq. Aks holda, qidiruv pichan oʻtidan igna topishga oʻxshaydi.

Barcha ish vaqtini saralash algoritmlarini oʻrganish va amalga oshirishga sarflaydigan dasturchilar mavjud. Buning sababi shundaki, biznesdagi dasturlarning aksariyati maʼlumotlar bazasini boshqarishni oʻz ichiga oladi. Odamlar doimo maʼlumot bazalarida maʼlumot qidirishadi. Bu shuni anglatadiki, qidirish algoritmlari juda mashhur.

Ammo bitta "lekin" bor. Qidiruv algoritmlari allaqachon saralangan maʼlumotlar bazalari bilan ancha tez ishlaydi. Bunday holda, faqat chiziqli qidirish kerak.

Baʼzi bir vaqtlarda kompyuterlar foydalanuvchilarsiz boʻlganda, saralash algoritmlari maʼlumotlar bazalari bilan ishlashda davom etmoqda. Qidiruv foydalanuvchilari yana keladi va maʼlumotlar bazasi maʼlum bir qidiruv maqsadiga qarab allaqachon saralanadi.

Tanlash (Selection) saralash algoritmi:

Massivni koʻtarilish tartibida tartiblash uchun har bir iteratsiyada eng yuqori qiymatga ega elementni topish kerak. Uning yordamida siz oxirgi elementni almashtirishingiz kerak. Eng yuqori qiymatga ega boʻlgan keyingi element

allaqachon oxirgi, ammo bitta joyda. Bu massivning birinchi joylaridagi elementlar kerakli tartibda bo‘lmaguncha sodir bo‘lishi kerak.

Dastur matni:

```
1. #include "stdafx.h"
2. #include<iostream>
3. using namespace std;
4. void selectionSort(int data[], int lenD){
5. int j = 0;
6. int tmp = 0;
7. for(int i=0; i<lenD; i++){
8. j = i;
9. for(int k = i; k<lenD; k++){
10. if(data[j]>data[k]){
11. j = k;}
12. }
13. tmp = data[i];
14. data[i] = data[j];
15. data[j] = tmp;
16. }}
17. int main(){
18. int A[]={11,-2,33,2,7,88,-9};
19. selectionSort(A,7);
20. for(int i=0; i<7; i++){
21. cout<<A[i]<<" ";}
22. getchar();
23.}
```

Datur natijasi:

-9 -2 2 7 11 33 88

Pufakchali (Bubble sort) saralash algoritmi:

Qabariqni saralash bilan qo‘shni elementlar taqqoslanadi va almashtiriladi, agar keyingi element avvalgisidan kichikroq bo‘lsa. Bir nechta ma’lumotlarni uzatish kerak. Birinchi o‘tish paytida massivdagi dastlabki ikkita element bir-biriga mos keladi. Agar ular tartibda bo‘lmasa, ular almashtiriladi va keyin keyingi juftlikdagi elementlar taqqoslanadi. Xuddi shu shartda ular joylarni o‘zgartiradilar. Shunday qilib, tartiblash har bir tsiklda massiv oxiriga etgunga qadar sodir bo‘ladi.

Dastur matni:

```
1. #include "stdafx.h"
2. #include<iostream>
3. using namespace std;
4. void bubbleSort(int data[], int lenD){
5. int tmp = 0;
6. for(int i = 0; i<lenD; i++){
7. for(int j = (lenD-1); j>=(i+1); j--){
8. if(data[j]<data[j-1]){
9. tmp = data[j];
10. data[j]=data[j-1];
11. data[j-1]=tmp;}}}}
12. int main(){
13. int A[]={11, -2, 33, 2, 7, 88, -9};
14. bubbleSort(A, 7);
15. for(int i=0; i<7; i++){
16. cout<<A[i]<<" ";
17. }
18. getchar();
19. }
```

Datur natijasi:

-9 -2 2 7 11 33 88

Qo'shish tartibida (Insertion sort) saralash algoritmi:

Qo'shimchalar bo'yicha tartiblashda, qator ikki qismga bo'linadi: tartiblangan va tartibsiz. Dastlab, butun massiv tartibga solinmagan maydon. Birinchi o'tish joyida tartibsiz hududning birinchi elementi olib tashlanadi va buyurtma qilingan hududda to'g'ri joyga joylashtiriladi.

Har bir o'tish joyida buyurtma qilingan mintaqaning o'lchami 1 ga oshadi va tartibsiz hududning hajmi 1 ga kamayadi.

Asosiy tsikl 1 dan N-1 gacha ishlaydi. Jth iteratsiyada [i] elementi tartibga solingan hududda to'g'ri joylashtirilgan. Bu [i] dan kattaroq buyurtma qilingan mintaqaning barcha elementlarini bitta pozitsiyani o'ngga siljitish orqali amalga

oshiriladi. [i] [i] dan kichikroq va [i] dan katta bo'lgan elementlar orasidagi oraliqqa kiritilgan.

Dastur matni:

```
1. #include "stdafx.h"
2. #include<iostream>
3. using namespace std;
4. void insertionSort(int data[], int lenD){
5. int key = 0;
6. int i = 0;
7. for(int j = 1;j<lenD;j++){
8. key = data[j]; i = j-1;
9. while(i>=0 && data[i]>key){
10. data[i+1] = data[i];
11. i = i-1;
12. data[i+1]=key;}} }
13. int main(){
14. int A[]={11,-2,33,2,7,88,-9};
15. insertionSort(A,7);
16. for(int i=0; i<7; i++){
17. cout<<A[i]<<" "; } getchar();
18. }
```

Datur natijasi:

-9 -2 2 7 11 33 88

Birlashtirish bo'yicha (Merge sort) saralash algoritmi:

Birlashtirish orqali rekursiv saralashda massiv dastlab kichik qismlarga bo'linadi - birinchi bosqichda - bitta elementdan iborat bo'lganlarga. Keyin bu qismlar kattaroq qismlarga birlashtiriladi - har biri ikkita element va elementlar taqqoslanadi va natijada yangi qismda kichikroq element chap tomonda, kattarog'i esa o'ngda bo'ladi. Keyinchalik kattaroq qismlarga birlashish jarayoni amalga oshiriladi va hokazo, barcha qismlar bitta qismga birlashtirilganda, allaqachon saralangan algoritm oxirigacha. Agar qiziqish bo'lsa, rekursiv funktsiyalar haqida maqola bor.

Dastur matni:

```
1. #include "stdafx.h"
2. #include<iostream>
3. using namespace std;
4. void merge(int merged[], int lenD, int L[], int lenL,
   int R[], int lenR){
5. int i = 0;
6. int j = 0;
7. while(i<lenL||j<lenR){
8. if (i<lenL & j<lenR){
9. if(L[i]<=R[j]){
10. merged[i+j] = L[i];
11. i++;}
12. else{
13. merged[i+j] = R[j];
14. j++;}
15. }
16. else if(i<lenL){
17. merged[i+j] = L[i];
18. i++;}
19. else if(j<lenR){
20. merged[i+j] = R[j];
21. j++;}}
22. }
23. void mergeSort(int data[], int lenD){
24. if(lenD>1){
25. int middle = lenD/2;
26. int rem = lenD-middle;
27. int* L = new int[middle];
28. int* R = new int[rem];
29. for(int i=0;i<lenD;i++){
30. if(i<middle){
31. L[i] = data[i];}
32. else{
33. R[i-middle] = data[i];}
34. }
35. mergeSort(L,middle);
36. mergeSort(R,rem);
37. merge(data, lenD, L, middle, R, rem);}
38. }
39. int main(){
40. int A[]={11,-2,33,2,7,88,-9};
```

```

41. mergeSort(A,7);
42. for(int i=0; i<7; i++){
43. cout<<A[i]<<" ";}
44. getchar();
45. }

```

Datur natijasi:

-9 -2 2 7 11 33 88

Tezkor (Quick sort) saralash algoritmi:

Quick Sort bo‘linish va tugatish algoritmidan foydalanadi. Dastlabki qatorni ikkita maydonga bo‘lish bilan boshlanadi. Ushbu qismlar mos yozuvlar deb nomlangan elementning chap va o‘ng tomonida joylashgan. Jarayon oxirida, bitta qism mos yozuvdan kichikroq elementlarni, boshqa qismi esa mos yozuvlar darajasidan kattaroq elementlarni o‘z ichiga oladi.

Dastur matni:

```

1. #include "stdafx.h"
2. #include<iostream>
3. using namespace std;
4. void quickSort(int* data, int const len)
5. {
6. int const lenD = len;
7. int pivot = 0;
8. int ind = lenD/2;
9. int i,j = 0,k = 0;
10. if(lenD>1){
11. int* L = new int[lenD];
12. int* R = new int[lenD];
13. pivot = data[ind];
14. for(i=0;i<lenD;i++){
15. if(i!=ind){
16. if(data[i]<pivot){
17. L[j] = data[i];
18. j++; }
19. else{
20. R[k] = data[i];
21. k++; } } }

```

```

22. quickSort(L,j);
23. quickSort(R,k);
24. for(int cnt=0;cnt<lenD;cnt++){
25. if(cnt<j){
26. data[cnt] = L[cnt];;
27. }
28. else if(cnt==j){
29. data[cnt] = pivot;
30. }
31. else{
32. data[cnt] = R[cnt-(j+1)]; }}}}
33. }
34. int main(){
35. int A[]={11,-2,33,2,7,88,-9};
36. quickSort(A,7);
37. for(int i=0; i<7; i++){
38. cout<<A[i]<<" ";}
39. getchar();
40. }

```

Datur natijasi:

-9 -2 2 7 11 33 88

Har xil tipdagi sonlar to‘plamini saralash algoritmi:

Ushbu algoritm shablon yordamida tuziladi. Shblonlar funksiyalarni ortiqcha qayta yuklanishdan saqlash uchun xizmat qiladi. Quyidagi **Universal_sort()** funksiyasi bunga misol bo‘ladi.

Dastur matni:

```

1. #include "stdafx.h"
2. #include<iostream>
3. #include<string.h>
4. template <class T>
5. T* merge_new(T *m1, T* m2, int l1, int l2){
6. T* ret = new T[l1+l2];
7. int n = 0;
8. while (l1 && l2){
9. if (*m1 < *m2){
10. ret[n] = *m1;

```

```

11. m1++; l1--;}
12. else {
13. ret[n] = *m2;
14. m2++; l2--;}
15. n++;}
16. if (l1 == 0){
17. for (int i=0; i<l2; i++){
18. ret[n++] = *m2++;}}
19. else {
20. for (int i=0; i<l1; i++){
21. ret[n++] = *m1++;}}
22. return ret;}
23. template <class T>
24. void Universal_sort(T * mas, int len){
25. int n=1, l, ost;
26. T * mas1;
27. while (n<len){
28. l=0;
29. while (l<len){
30. if (l+n >= len) break;
31. ost = (l+n*2>len) ? (len-(l+n)) : n;
32. mas1 = merge_new(mas+l, mas+l+n, n, ost);
33. for (int i=0; i<n+ost; i++) mas[l+i] = mas1[i];
34. delete [] mas1;
35. l+=n*2;}
36. n*=2;
37. }}
38. int main(){
39. int a[] = {0,-1,23,-5,7,9,71};
40. Universal_sort(a, 7);
41. cout<<"int turidagi to'plam"<<endl;
42. for(int i = 0; i < 7; ++i) {
43. cout << a[i] << " ";
44. }
45. float b[] = {5.1, 9.7, 11.3, -4.7, 88, -7.1, 88.6};
46. Universal_sort(b, 7);
47. cout<<endl<<"float turidagi to'plam"<<endl;
48. for(int i = 0; i < 7; ++i) {
49. cout << b[i] << " ";
50. }
51. char c[] = "Salom dunyo";
52. Universal_sort(c, strlen(c));

```

```

53. cout<<endl<<"char turidagi to'plam"<<endl;
54. cout<<<<endl;
55. getchar();
56.}

```

Datur natijasi:

int turidagi to'plam

-5 -1 0 7 9 23 71

float turidagi to'plam

-7.1 -4.7 5.1 9.7 11.3 88 88.6

char turidagi to'plam


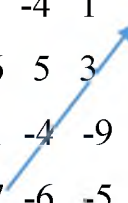
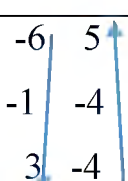
Sadlmnoouy

Laboratoriya mashg'ulotini bajarish bo'yicha variantlar

Yuqorida keltirilgan saralash algoritmlari asosida quyidagi jadvalda tasodifiy sonlar bilan to'ldirilgan ko'p o'lchovli massivlarni ko'rsatilganday qilib saralang:

8- jadval

№	To'plamni berilishi	Natijasi
1.	N=4 M=4 3 4 1 2 6 5 7 8 11 10 9 12 13 16 15 14	<u>1 2 3 4</u> → 5 6 7 8 9 10 11 12 13 14 15 16
2.	N=4 M=4 3 -4 1 -2 -6 5 -7 8 -1 -4 9 12 3 -6 -5 14	← <u>3 1 -2 -4</u> 5 8 -6 -7 12 9 -4 -1 14 3 -5 -6

3.	N=4 M=4 3 -4 1 24 -6 5 -7 18 -1 -4 9 12 3 -6 -5 -2	-6 -6 -7 -2 -1 -4 1 12 3 -4 5 18 3 5 9 24 
4.	N=4 M=4 3 -4 1 24 -6 5 -7 18 -1 -4 -9 12 3 -6 -5 -2	-9 -4 1 24 -6 -2 -7 18 -1 -4 3 12 3 -6 -5 5
5.	N=4 M=4 3 -4 1 24 -6 5 -7 18 -1 -4 -9 12 3 -6 -5 -2	3 -4 1 24 -6 5 3 18 -1 -4 -9 12 -7 -6 -5 -2 
6.	N=4 M=4 3 4 1 2 6 5 7 8 11 10 9 12 13 16 15 14	<u>1 2 3 4</u> <u>8 7 6 5</u> 9 10 11 12 16 15 14 13
7.	N=4 M=4 3 -4 1 24	-6 5 -7 24 -1 -4 1 18 3 -4 5 12 

	-6 5 -7 18 -1 -4 9 12 3 -6 -5 -6	3 -6 9 -6
8.	N=4 M=4 3 -4 1 24 -6 5 -7 18 -1 -4 9 12 3 -6 -5 -6	24 18 12 -4 9 5 3 3 -4 -6 -6 -6 1 -1 -5 -7
9.	N=4 M=4 3 -4 1 24 -6 5 -7 18 -1 -4 9 12 3 -6 -5 -6	24 9 -4 1 18 5 -6 -1 12 3 -6 -5 -4 3 -6 -7
10.	N=4 M=4 3 -4 1 24 -6 5 -7 18 -1 -4 9 12 3 -6 -5 -6	24 -4 9 1 18 -6 5 -1 12 -6 3 -5 -4 -6 3 -7
11.	N=4 M=4 3 -4 1 24 -6 5 -7 18 -1 -4 9 12 3 -6 -5 -6	24 18 12 -4 -4 -6 -6 -6 9 5 3 3 1 -1 -5 -7

12.	N=4 M=4 3 4	3 4 7 11 18 29 47 76 123 199 222 401 623 1024 1647 2671
13.	N=5 M=4 3 4 1 24 6 5 7 18	3 4 1 5 6 5 7 9 9 9 8 14 15 14 15 23 24 23 23 37
14.	N=3 M=4 3 4 6 5 7 18	3 4 7 11 6 5 11 16 7 18 25 43
15.	N = 4 1) Salimov 2) Odilov 3) Abdullayev 4) Afzalov	1) Abdullayev 2) Afzalov 3) Odilov 4) Salimov
16.	N = 4 1) Salimov 4 2) Odilov 3 3) Abdullayev 5 4) Afzalov 2	1) Abdullayev 5 2) Salimov 4 3) Odilov 3 4) Afzalov 2

Takrorlash uchun savla va topshiriqlar

1. Katta sonlar bilan ishlashda qanday tiplar qo‘llaniladi?
2. Katta sonlar bilan ishlovchi kutubxonalarni aniqlang.
3. Saralash algoritmlari va ularning farqli tomalarini aniqlang.
4. Katta sonlar uchun ham saralash algoritmlarini yarating.
5. Katta sonlar ustida matematik amallar bajaruvchi funksiyalar yarating.

6- LABORATORIYA ISHI

6. VISUAL C++ MUHITIDA DASTURLASH

6.1. Visual C++ muhitida menyular va uskunalar paneli.

Hozirda ko‘plab dasturlash tillari mavjud bo‘lib, ular qo‘llanilish sohasiga qarab turlicha bo‘ladi, ya‘ni har bir soha uchun mo‘ljallangan dasturlash tillari mavjud. Ularning bir nechtasini sanab o‘tish mumkin. Masalan, **C#, C++, Visual Basic, JavaScript, Delphi** va boshqalar.

Quyida ushbu dasturlash tillari bilan tanishib chiqamiz:

Visual Basic .NET - Microsoft Visual Studio 2016 tarkibidagi samaradorligi katta dasturlash tillardan biri bo‘lib, bu dasturlash tilini to‘liq obyektga yo‘naltirilgan dasturlash tili deb aytishimiz mumkin. **Visual Basic .NET** dasturlash tili yordamida **Windows** ilovalarini va Web ilovalarni yaratish mumkin.

Visual C# .NET - Microsoft korporasiyasi tomonidan ishlab chiqilgan bo‘lib, bu dasturlash tili aynan .NET platformasi uchun ishlab chiqilgan. **Visual C# .NET** dasturlash tili imkoniyatlari boshqa obyektga yo‘naltirilgan dasturlash tillari (**C, C++, Java va Delphi**)dan ancha keng bo‘lib, bu dasturlash tilida **Visual Basic .NET** kabi **Windows** ilovalarini va Web ilovalarni yaratish mumkin.

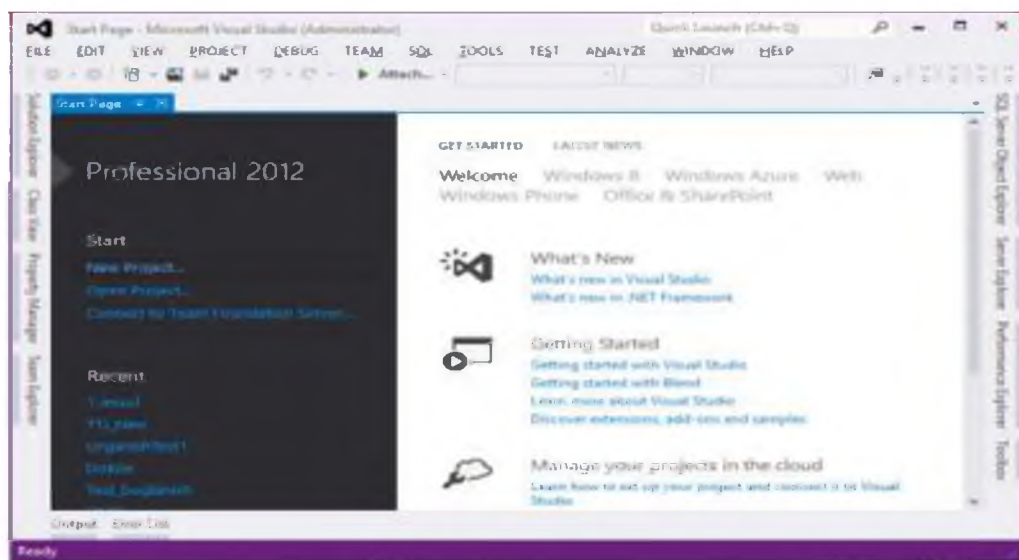
Visual C++ .NET dasturlash tili past sathdagi dasturchi uchun ilovalarni boshqarishda talab qilinadi. .NET platformasining **Visual C++** dasturini boshqa dasturlardan shu bilan farq qiladiki, bu dasturlash tili .NET platformasining kodli modeli (**managed code model**) va **Windows (unmanaged native code model)** kodli modelini qo‘llab quvvatlaydi.

Visual J# .NET - Microsoft .NET platformasi uchun Web-servis va ilovalar yaratuvchi Java dasturchilari ishlatishi mumkin.

Visual C#.NET dasturlash tili. Zamonaviy dasturlash tillari orasida **Visual C#.NET** mukammal dasturlash tillaridan biri hisoblanadi. Ushbu dasturlash tili C++

asosida kelib chiqqan bo‘lib, u .NET Microsoft platformasi uchun yaratilgan va obyektga yo‘naltirilgan dasturlashga mo‘ljallangan. Uning yordamida internetda juda ommabop sahifalar, saytlar yaratsa bo‘ladi va yaratilmoqda. **Visual C#.NET** dasturlash tilining afzalligi bu boshqa dasturlash tillarida yo‘l qo‘yilgan kamchiliklardan xoliligidir.

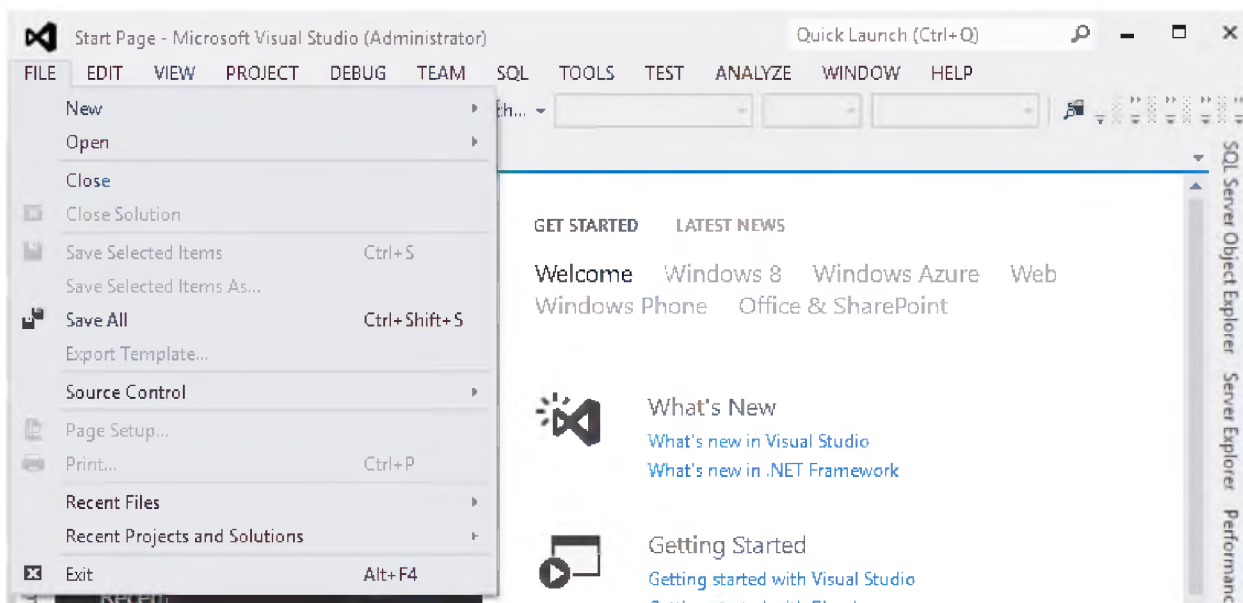
C++ tili C tilining rivojlangan holatidir. Shuning uchun C tilining barcha konstruksiyalari C++ dasturlash tilida o‘z ifodasini topadi. Biroq C++ tilida C tiliga qaraganda juda ko‘p sintaktik imkoniyatlar paydo bo‘ldi (biz buni material bilan tanishish jarayonida ko‘rib chiqamiz). Ushbu tilda dasturlash uchun biz **Visual Studio 2016** (ingliz tilidagi abriviaturasi – **IDE: Integrated Development Environment**) dasturi bilan yaqindan tanishishimiz zarur bo‘ladi. Shu sababli dastur muhitining strukturasi va undagi interfeys bilan yaqindan tanishamiz. Interfeys – bu shunday qurilmaki, bunda foydalanuvchi muhit bilan muloqot jarayoni osonlashadi. **Visual Studio 2016** dasturini yuklab olib, uni o‘rnatganingizdan so‘ng -> **Пуск | Программы** paneliga kirib, uning .exe faylini topishingiz mumkin.



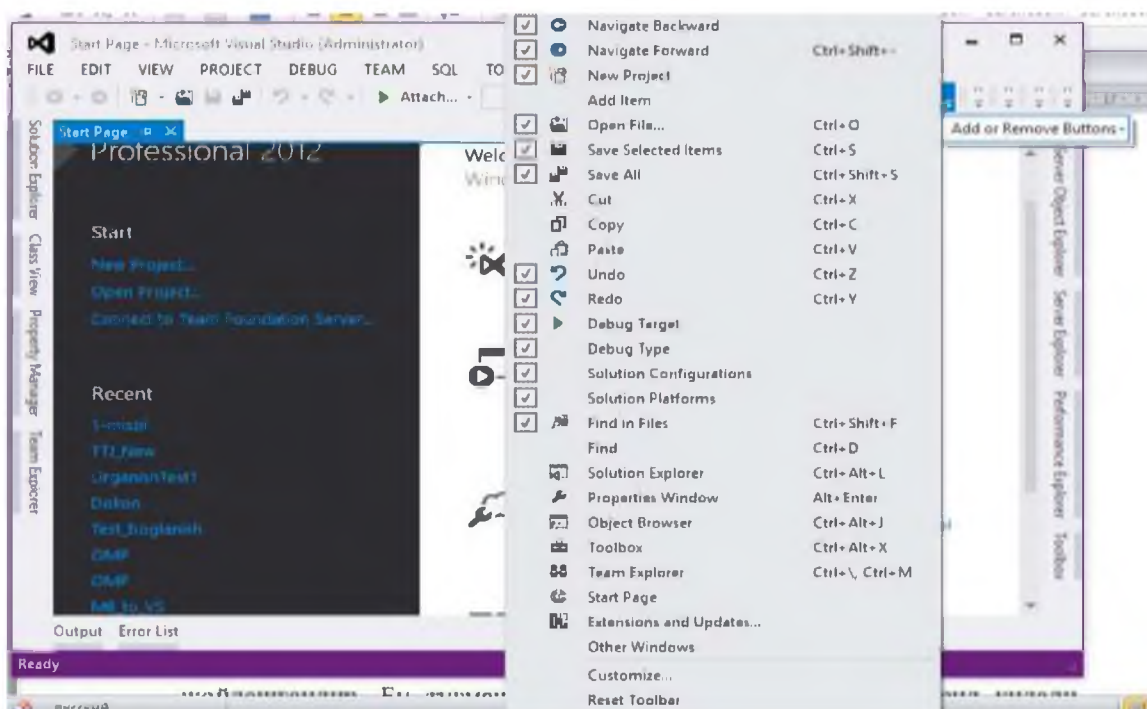
6.1- rasm. Visual Studio 2012 IDE oynasini ko‘rinishi

Yuqori qismda **File**, **Edit** tugmalari joylashgan bo‘lib – bu gorizontol menyu hisoblanadi. Ushbu buyruqlarni chaqirish imkoniyatlari juda keng bo‘lib, “tushib

qoluvchi” oynalar - vertikal menyu hisoblanadi hamda oynaning chap yuqori qismidan boshlab joylashadi va o‘zida turli xil buyruqlar majmuasini saqlaydi.



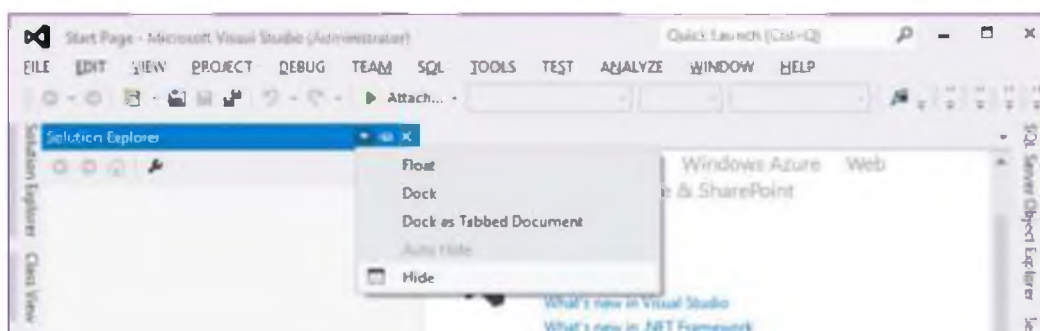
6.2- rasm. Hosil qilinayotgan ilova menyusi



6.3- rasm. Tezkor tugmalar majmuasi

Pastki satrlarda asosiy menyuda **tezkor murojaatlar oynasi** joylashgandir. Bu tugmachalar suzib chiquvchi oynalarni hosil qiladi. Uning yonida kerakli qo‘shimcha tugmachalar ham joylashgandir. Asosiy oyna ko‘rinishi ilova turiga qarab o‘z holatini o‘zgartiradi.

Ishchi oyna oynalar majmuasidan iboratdir. Har bir oyna – bu odatda **windows** – oyna bo‘lib, u standart sarlavha qatoriga egadir. Bunda sichqoncha orqali sarlavha qatorlarini ixtiyoriy joyga surish mumkin. 4.4 - rasmda **Solution Explorer** oynasi ko‘rsatib o‘tilgan.



6.4- rasm. Oynalar majmuasi tarkibi

Ba`zi bir tarkibiy tuzilmalarni ko‘rib chiqamiz:

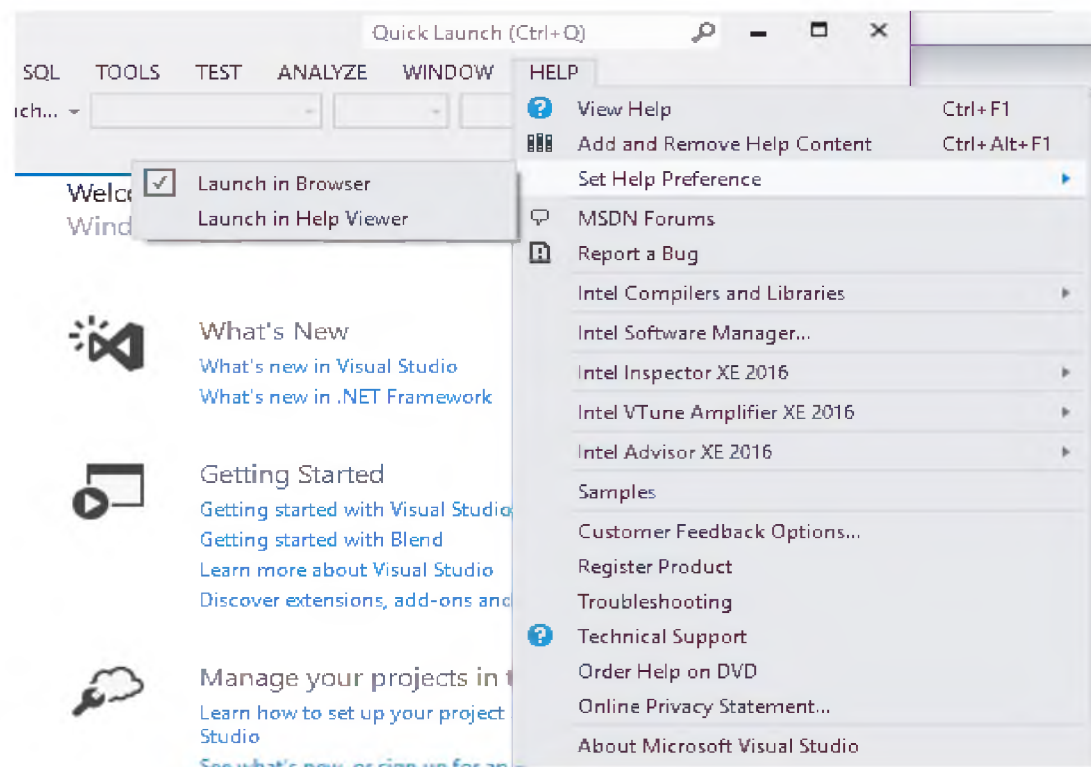
- **Float** (suzuvchi). Ushbu oynani ishchi oynada turli joyga cho‘zish mumkin.
- **Dock** (yondashuvchi). Ushbu oyna joyini o‘zgartirish mumkin (boshqa oynalar orqali fiksirlanadi).
- **Dock as Tabbed Document** (yopishgan oyna holatida asosiy oynada bo‘ladi). Bu oynada **Start page** varaqasi mavjud bo‘ladi.
- **Auto Hide** (avtomatik tarzda yo‘qolib qolish). Ushbu holatda oyna avtomatik tarzda yo‘qoladi va ishchi oynaning yon tomoniga yopishib turadi va sichqoncha olib borilganda u birdagina suzib chiqadi.

- **Hide** (yashirish). Bu holatda oyna ekrandan butunlay yo‘qoladi. Uni qaytarish uchun asosiy menyudan **View, Other Windows** parametridan foydalanish zarur bo‘ladi.

Oynalar ustida biror – bir operasialarni bajarish uchun ularning tarkibini yaxshi bilish kerak bo‘ladi.

Misol uchun Toolbox va **Server Explorer** ma‘lumot-oynasini qanday qilib o‘rnatish? Ushbu oynalarda tarkibini o‘rnatish (sichqonchanning o‘ng tomonini bosish orqali). Shunda ekranda **Auto Hide** (avtomatik berkinuvchi) chiqadi va shunga o‘xshash tugmachalarni ishlatish mumkin.

Help yordamchi tizimi haqida ma‘lumot



6.5- rasm. Help menyusi

Hammani quyidagi savol qiziqtirishi tabiiy: “Bu qanday amalga oshiriladi?”. Ma‘lumki, yordamchi ma‘lumotlar oynasi tizim dasturida mavjud. Buning uchun **Help** tugmasi tanlanadi (MV Visual C++ 4.5 - rasm)

Visual C++ muhitining strukturasi

Visual C++ dasturlash muhitidagi dasturlar ilovalar deyiladi. (ko‘rinib turibdiki, IDE dasturlash muhiti). Biz ularni kelgusida Visual C++ ilovalari deb ataymiz. Ushbu ilovalar konstruksiya ko‘rinishida tuziladi va ular - loyihalar deyiladi. Bu bir necha fayllarning ketma- ket jamlanmasi hisoblanadi. C++ dasturlash tilidagi dasturlar bu funksiyalar jamlanmasi hisoblanib, kerakli majburiyatlarga ega bo‘ladi. Ilova – bu asosiy funksiya bo‘lib, uning ichida shunday operatorlar mavjudki, u dastur ishlashini realizatsiya qiladi. Har bir dastur o‘z ishini asosiy funksiyadan boshlaydi va bu funksiyaning bir qismi dasturchi tomonidan tuziladi, qolgan qismi esa – kutubxona, ya‘ni sarlavha funksiyalari tomonidan dasturlash jarayonida foydalanuvchiga uzatiladi. C/C++ tilini o‘rganishda biz maxsus ilovalar turlaridan foydalanamiz – konsol rejimidagi ilovalarda ishlash, ya‘ni oldindan hosil qilingan shablonlar asosida hosil qilinadigan holatlar bilan tanishamiz.

Konsol oynasi – bu grafik interfeysga ega bo‘lmagan, dastur bilan foydalanuvchi orasidagi buyruqlar oynasi orqali hosil qilinadigan natija oynasidir. Buning uchun biz birinchi dialoglar oynasidan **File | New | Project** buyrug‘ini tanlaymiz. Loyihaning yig‘ilish va kompilyatsiya jarayoni **Build** buyrug‘i orqali amalga oshiriladi. Kompilyatsiya jarayonidan so‘ng **Debug** buyrug‘i orqali dasturni bajarish jarayoniga o‘tiladi.

VC++ tilini o‘rganishni biz turli misollarda ko‘rib chiqamiz, turli dasturlar hosil qilamiz, ularni tahlil qilamiz, va ularni ishlash strukturalari bilan parallel ravishda tanishib boramiz. Konsol rejimidagi shablonlarni biz CLR (Common Language Runtime) **Console Application** da hosil qilamiz. Oxirgi ikki so‘z konsol ilova deganidir. **CLR** nima degan savol tug‘ilishi tabiiydir. **Visual C++ 2008** dasturlash muhitida konsol ilovalarining 2 ta shablони mavjud: ulardan birinchisi – biz ko‘rib chiqayotgan shablon va ikkinchisi – abriviaturasiz shablon hisoblanadi. **CLR** – bu maxsus muhit bo‘lib, dastur kodining bajarilishini boshqaradi hamda

dastur kodining xavfsizligi va to'g'ri bajarilishini ta'minlaydi. 2005 va 2008 yilgi **Visual C++** dasturlash versiyalarida ushbu jarayon kompilyatsiyada bajarilar edi. Bunda dastur uchun "to'plam" degan xotira ajratilar edi: unda biz obyektни joylashtirish, xotirani bo'shatish, obyekt bilan ishlash vaqti kabi parametrlarni bajarar edik. Boshqa tomondan, **CLR** rejimi yoqilganda, bu ilova boshqa ilovalarga nisbatan tubdan farq qiladi, bunda ilovaga ulanish **System** muhiti orqali amalga oshiriladi, bunda obyektning xotiraga joylashishi avtomatik boshqariladi.

Regulyatsiyalanuvchi ko'rsatkich – bu shunday ko'rsatkich turiki, bunda havola orqali obyektning "to'plam" xotiradagi o'rni ko'rsatiladi, bu holat sinov paytida amalga oshiriladi. Bunday ko'rsatkichlar uchun "*" o'rniga "^" belgisi ishlatiladi. Xullas, **CLR** o'zgaruvchilarni hosil qilish apparatini, ya'ni bir to'plamga tegishli xotira adresi va h.k. yaratib beradi. Shunday maxsus kutubxona borki, bu jarayonni boshqaradi. Lekin bu narsalar dasturlashni juda qiyinlashtirib yuboradi.

Biz bu bo'limda muhitning bosh oynasini hamda loyihalarda ishlatiladigan asosiy formalar va ularning sifatli chiqishi uchun zarur bo'lgan maxsus konstruksiyalarni qarab chiqamiz. Bu yerda ham konsolli ilovalarda o'rganilgan tushunchalardan foydalaniladi. Bu va bundan keyingi bo'limlarda biz murakkab ilovalarning grafik interfeysi bilan tanishamiz. Ilovalarni yaratishda forma tushunchasini ishlatamiz. Bosh oynaning chap qismida ikki bo'lakli vkladka joylashgan. Birinchi qism ma'lumotlar bazasi bilan ishlashni tashkil etsa, ikkinchi qismda komponentalar ro'yxati keltirilgan.

Bosh oyna (ishchi stol) ning har bir qismi to'g'risidagi to'liqroq ma'lumotlarni biz keyingi boblarimiz davomida ko'rib chiqamiz.

Agar ishchi stolimizning strukturasi e'tibor bersak, har bir oynaning sarlavhasi o'z tarkibida bo'lgan ko'plab funksional masalalarga mos ravishda tanlangan. Bu oynalarni sichqonchaning chap tugmasi yordamida istalgan joyga joylashtirish mumkin. Oynalarni o'zaro birlashtirib, bir nechta vkladkalar ko'rinishida ham joylashtirsa bo'ladi. Siz agar biror bir oynani tanlab, sichqoncha yordamida kerakli

joyga oʻrnatmoqchi boʻlsangiz, darhol sizga kerakli obyektни tanlash imkonini beruvchi chizmalar beriladi. Siz ulardan birini tanlab, kerakli joyga oynani oʻrnatishingiz mumkin.

Demak, birinchi ish muhitning oynalarini oʻzingizga qulay tarzda oʻrnatishdir. Aslida bu ish oddiy koʻringani bilan, lekin keyinchalik ish unumdorligiga juda katta taʼsir qiladi, yaʼni vaqtdan yutish imkonini beradi. Loyihalaringizni tayyorlayotganda barcha oynalar sizning qoʻl ostingizda qulay tarzda joylashgan boʻlsa, ish sifati ham oshadi.

Bu oynalar bilan ishlash davomida goʻyoki mashhur dasturlardan biri boʻlmish Photoshop oynalari yodga keladi. Bu oynalarning xossalari bir-biri bilan uzviy bogʻliq. Oynalarni keraklicha joylashtirishni muhitning *Bud* menyusi orqali ham amalga oshirish mumkin.

Ishning maksadi

Visual C++ muhitida **Windows Application** ilova dasturlar yaratish usullari bilan tanishish.

Topshiriq

Maʼruza va amaliyot darslarida olingan maʼlumotlarga asoslanib, oʻz variantingiz uchun quyidagilarni bajarinig.

- a) 9- jadvaldan oʻz variantini koʻchirib oling;
- b) quyidagi namuna asosida oʻz variantingizdagi masalalarni dasturi tuzing.

9- jadval

<u>Varianlar</u>
1. $a[n][m]$ massiv elementlarini $[0; 10000]$ oraliqdagi sonlar bilan toʻldiring. Bu massivning har bir satr elementlarini oʻsish tartibida chiqaring.

2. $a[n][m]$ massiv elementlarini $[0; 10000]$ oraliqdagi sonlar bilan to'ldiring. Uning eng katta va eng kichik elementini toping. Agarda bunday elementlar ko'p bo'lsa, ularning sonini chiqaring.
3. $a[n][m]$ massiv elementlarini $[-1000; 1000]$ oraliqdagi tasodifiy sonlar bilan to'ldiring. Uning eng katta va eng kichik elementlarini nomer(indeks)larini toping.
4. $a[n][m]$ massiv elementlarini $[-10000; 10000]$ oraliqdagi tasodifiy sonlar bilan to'ldiring. Uning eng katta va eng kichik elementlarining ayirmasini toping.
5. $a[n][m]$ massiv elementlarini $[-500; 500]$ oraliqdagi tasodifiy sonlar bilan to'ldiring. Bu massivni ustun elementlarini o'sish tartibida chiqaring.
6. $a[n][m]$ massiv elementlarini $[-900; 900]$ oraliqdagi tasodifiy sonlar bilan to'ldiring. Bu massivni bosh diagonal elementlarini o'sish tartibida chiqaring. Albatta mos ravishda satrlari ham o'zgarishi lozim
7. $a[n][m]$ massiv elementlarini $[0; 100000]$ oraliqdagi tasodifiy sonlar bilan to'ldiring. Elementining qiymati 200 dan kattalari sonini va ularning o'rta arifmetigini toping.
8. $a[n][m]$ massiv elementlarini $[-1000; 1000]$ oraliqdagi tasodifiy sonlar bilan to'ldiring. Bu massivning bosh diagonal elementlarini kamayish tartibida chiqaring.
9. $a[n][m]$ massiv elementlarini $[-100000; 100000]$ oraliqdagi tasodifiy sonlar bilan to'ldiring. Elementining qiymatini bir vaqtda 5 ga va 3 ga qoldiqsiz bo'linadiganlarining soni va ularni yig'indisini toping.
10. $a[n][m]$ massiv elementlarini $[-10000; 10000]$ oraliqdagi tasodifiy sonlar bilan to'ldiring. Massivni chap diagonal elementlarini kamayish tartibida chiqaring.
11. $a[n][m]$ massiv elementlarini $[-10000; 10000]$ oraliqdagi tasodifiy sonlar bilan to'ldiring. Massiv elementlari tublari sonini va ularning yig'indisini chiqaring.
12. $a[n][m]$ massiv elementlarini $[-10000; 10000]$ oraliqdagi tasodifiy sonlar bilan to'ldiring. Massivning eng katta va eng kichik elementlari sonini chiqaring.
13. $a[n][m]$ massiv elementlarini $[-1000; 1000]$ oraliqdagi tasodifiy sonlar bilan to'ldiring. Massivning eng oxirgi manfiy elementining nomerini aniqlang va chiqaring.

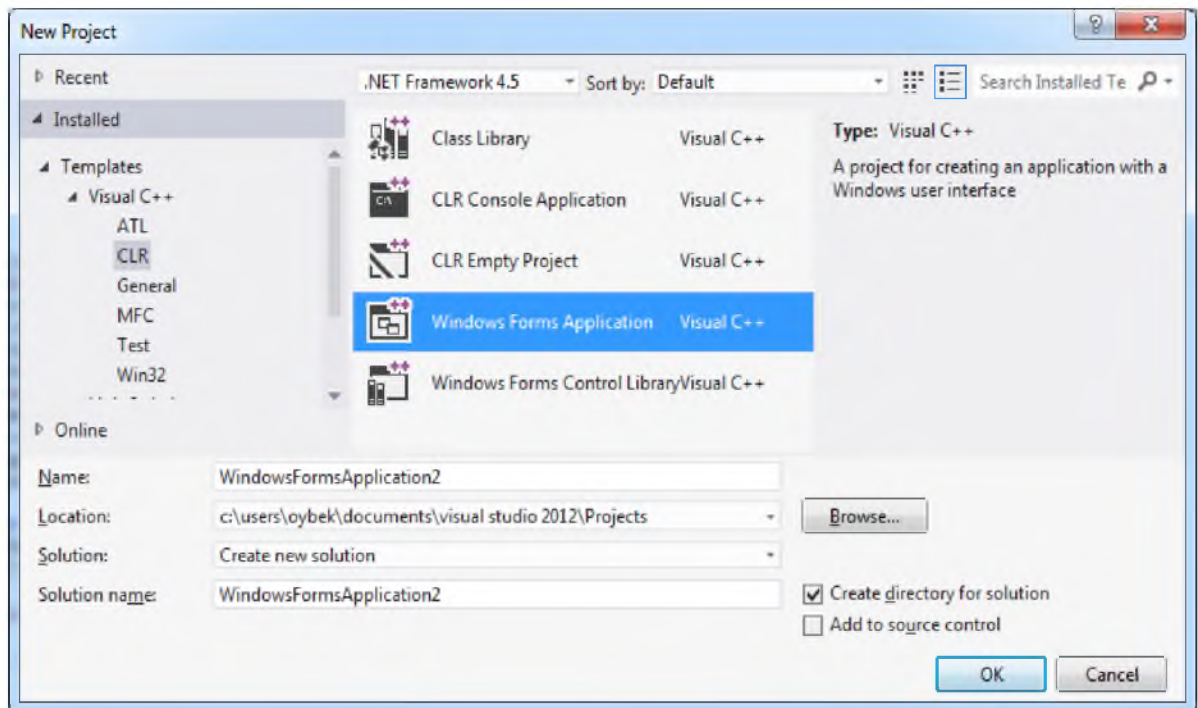
14.a[n][m] massiv elementlarini [-1000; 1000] oraliqdagi tasodifiy sonlar bilan to'ldiring. Massivning eng oxirgi musbat elementining nomerini aniqlang va chiqaring.
15.a[n][m] massiv elementlarini [-1000; 1000] oraliqdagi tasodifiy sonlar bilan to'ldiring. Massivning eng oxirgi manfiy elementining nomerini aniqlang va chiqaring.
16.a[n][m] massiv elementlarini [-1000000; 100000] oraliqdagi tasodifiy sonlar bilan to'ldiring. Massivning musbat va manfiy elementlarining yig'indisini toping.
17.a[n][m] massiv elementlarini [-1000; 1000] oraliqdagi tasodifiy sonlar bilan to'ldiring. Massivning eng oxirgi manfiy elementining nomerini aniqlang va chiqaring.
18.a[n][m] massiv elementlarini [-10000; 10000] oraliqdagi tasodifiy sonlar bilan to'ldiring. Massivning eng oxirgi manfiy elementining nomerini aniqlang va chiqaring.
19.a[n][m] massiv elementlarini [-1000; 1000] oraliqdagi tasodifiy sonlar bilan to'ldiring. Berilgan massiv elementlarining a va b oraliqdagilari sonini toping.
20.a[n][m] massiv elementlarini [-1000; 1000] oraliqdagi tasodifiy sonlar bilan to'ldiring. Massivni manfiy elementlarini ularning absolyut qiymati bilan almashtiring va almashtirishlar sonini chiqaring.
21.a[n][m] massiv elementlarini [-1000; 1000] oraliqdagi tasodifiy sonlar bilan to'ldiring. Massivning manfiy elementlaridan yangi matrisani hosil qiling.
22.a[n][m] massiv elementlarini [-1000; 1000] oraliqdagi tasodifiy sonlar bilan to'ldiring. Bu massiv elementlaridan 3 ga va 11 ga bo'linadiganlarining yig'indisi toping.
23.a[n][m] massiv elementlarini [-1000; 1000] oraliqdagi tasodifiy sonlar bilan to'ldiring. Massiv elementlarining o'rta arifmetigini toping.
24.a[n][m] massiv elementlarini [-1000; 1000] oraliqdagi tasodifiy sonlar bilan to'ldiring. Bu massiv elementlaridan 3 ga bo'linadiganlarining o'rta arifmetigi va o'rta geometrigini toping.

25.a[n][m] massiv elementlarini [-1000; 1000] oraliqdagi tasodifiy sonlar bilan to'ldiring. Massivni elementlaridan musbatlarini alohida S massivga va manfiylarini V massivga chiqaring.
26.a[n][m] massiv elementlarini [-5874; 10000] oraliqdagi tasodifiy sonlar bilan to'ldiring. Massivni elementlarini teskari tartibda chiqaring.
27.a[n][m] massiv elementlarini [-10000; 10000] oraliqdagi tasodifiy sonlar bilan to'ldiring. Massiv elementlarining birinchi musbatlarini keyin manfiylarini chiqaring.
28.a[n][m] massiv elementlarini [-10000; 100000] oraliqdagi tasodifiy sonlar bilan to'ldiring. Massivni eng katta va eng kichik elementlari orasidagi elementlarni chiqaring.
29.a[n][m] massiv elementlarini [-100000; 100000] oraliqdagi tasodifiy sonlar bilan to'ldiring. Massivning eng katta va eng kichik elementlari orasida bo'lmagan elementlarni chiqaring.
30.a[n][m] massiv elementlarini [-1000; 1000] oraliqdagi tasodifiy sonlar bilan to'ldiring. Massivning 7 ga bo'linadigan elementlarini massivdan o'chirib o'ringa 0 yozing.
31.a[n][m] massiv elementlarini [-897487; 1000] oraliqdagi tasodifiy sonlar bilan to'ldiring. Massiv elementlaridan qiymati toq bo'lganlarining yig'indisini toping.
32.a[n][m] massiv elementlarini [-10000; 10000] oraliqdagi tasodifiy sonlar bilan to'ldiring. Massiv elementlaridan qiymati juft bo'lganlarining yig'indisini toping.

9- jadvaldagi topshiriqlarni bajarish uchun uslubiy qo'llanmalar

Visual C++ dasturida 32–variantdagi masalani Windows Application muhitida ishlanishi:

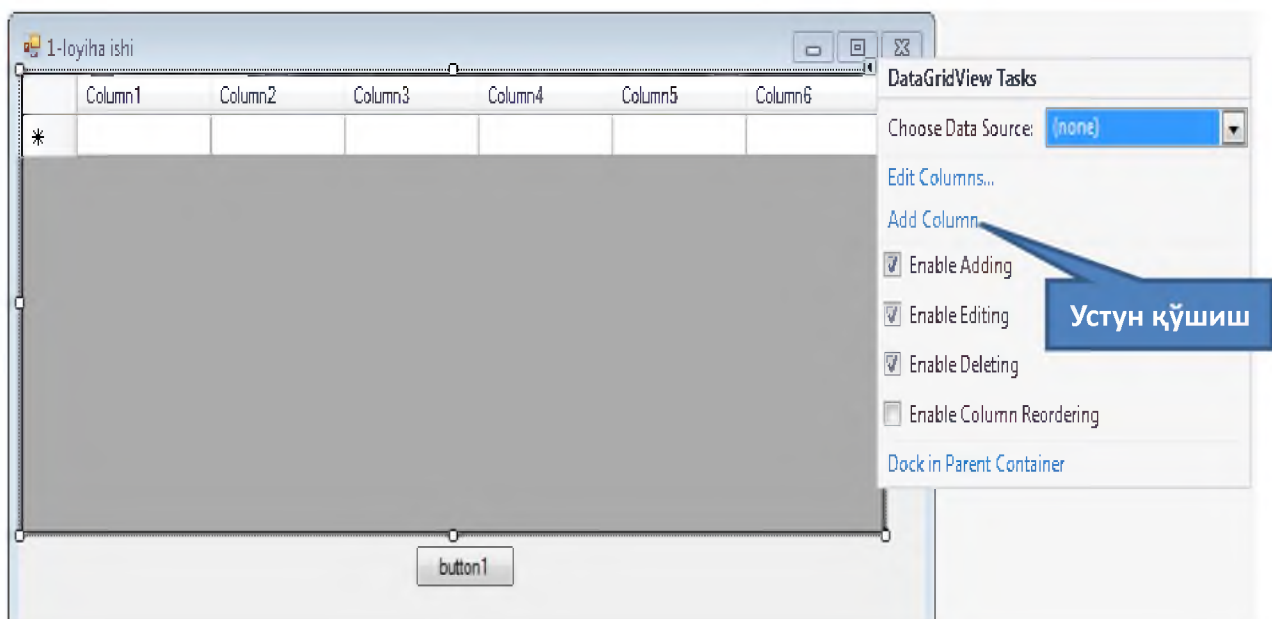
1-qadam.



6.6- rasm. Yangi loyiha turini tanlash oynasi

2- qadam.

Toolbox dialog oynasidan **Button**, **dataGridView**, **label** komponentalari formaga tashlanadi va **dataGridView** ga ustunlar qo‘shiladi. **dataGridView** ga vizual muhitdan ustunlar qo‘shsa bo‘ladi. Ustunlar qo‘shish quyidagicha:



6.7- rasm. dataGridView komponentasiga ustun qo‘shish

3-qadam.

Button tugmasini ustida sichqonchani chap tugmasi 2 marta tez bosiladi va kodlar oynasiga o'tiladi. Kodlar oynasida quyidagi kodlar yoziladi:

```
1. private: System::Void button1_Click(System::Object^
   sender, System::EventArgs^ e)
2. {
3. int n=Convert::ToInt32(textBox1->Text);
4. int s=0;
5. dataGridView1->ColumnCount = n;
6. dataGridView1->RowCount = n;
7. int a[10][10];
8. for(int i=0;i<n;i++){
9. for(int j=0;j<n;j++){
10. a[i][j]=rand()%50+1;
11. if(a[i][j]%2==0)s+=a[i][j];
12. dataGridView1->Columns[j]->HeaderText=
   (j+1).ToString()+" - ustun";
13. dataGridView1->Rows[i]->Cells[j]->Value
   =a[i][j].ToString();
14. }}
15. MessageBox::Show(Convert::ToString(s));
16. }
```

Visual C++ da tashqi ko'rinishga ega bo'lgan dasturlar yaratilganligi uchun komponentalarga avtomatik kodlar hosil qiladi. Dasturning to'liq kodi quyidagicha:

```
1. #pragma once
2. #include<math.h>
3. #include<stdlib.h>
4. namespace DataGridView_komponentasi {
5. using namespace System;
6. using namespace System::ComponentModel;
7. using namespace System::Collections;
8. using namespace System::Windows::Forms;
9. using namespace System::Data;
10. using namespace System::Drawing;
11. /// <summary>
12. /// Summary for Form1
13. /// </summary>
```

```

14. public ref class Form1 : public
    System::Windows::Forms::Form
15. {
16. public:
17. Form1(void){
18. InitializeComponent();
19. //
20. //TODO: Add the constructor code here
21. //
22. }
23. protected:
24. /// <summary>
25. /// Clean up any resources being used.
26. /// </summary>
27. ~Form1(){
28. if (components){
29. delete components;}
30. }
31. private: System::Windows::Forms::DataGridView^
    dataGridView1;
32. protected:
33. private: System::Windows::Forms::TextBox^ textBox1;
34. private: System::Windows::Forms::Label^ label1;
35. private: System::Windows::Forms::Button^ button1;
36. private:
37. /// <summary>
38. /// Required designer variable.
39. /// </summary>
40. System::ComponentModel::Container ^components;

41. #pragma region Windows Form Designer generated code
42. /// <summary>
43. /// Required method for Designer support - do not
    modify
44. /// the contents of this method with the code editor.
45. /// </summary>
46. void InitializeComponent(void){
47. this->dataGridView1 = (gcnew
    System::Windows::Forms::DataGridView());
48. this->textBox1 = (gcnew
    System::Windows::Forms::TextBox());

```



```

49. this->label1 = (gcnew
    System::Windows::Forms::Label());
50. this->button1 = (gcnew
    System::Windows::Forms::Button());
51. (cli::safe_cast<System::ComponentModel::ISupportInitia
    lize^ >(this->dataGridView1))->BeginInit();
52. this->SuspendLayout();
53. //
54. // dataGridView1
55. //
56. this->dataGridView1->ColumnHeadersHeightSizeMode =
    System::Windows::Forms::DataGridViewColumnHeadersHeightS
    izeMode::AutoSize;
57. this->dataGridView1->Location =
    System::Drawing::Point(12, 44);
58. this->dataGridView1->Name = L"dataGridView1";
59. this->dataGridView1->Size = System::Drawing::Size(584,
    223);
60. this->dataGridView1->TabIndex = 0;
61. //
62. // textBox1
63. //
64. this->textBox1->Font = (gcnew
    System::Drawing::Font(L"Microsoft Sans Serif", 14.25F,
    System::Drawing::FontStyle::Regular,
    System::Drawing::GraphicsUnit::Point,
65. static_cast<System::Byte>(204)));
66. this->textBox1->Location = System::Drawing::Point(93,
    12);
67. this->textBox1->Multiline = true;
68. this->textBox1->Name = L"textBox1";
69. this->textBox1->Size = System::Drawing::Size(100, 26);
70. this->textBox1->TabIndex = 1;
71. // label1
72. this->label1->AutoSize = true;
73. this->label1->Font = (gcnew
    System::Drawing::Font(L"Microsoft Sans Serif", 14.25F,
    System::Drawing::FontStyle::Regular,
    System::Drawing::GraphicsUnit::Point,
74. static_cast<System::Byte>(204)));
75. this->label1->Location = System::Drawing::Point(61,
    12);

```

```

76. this->label1->Name = L"label1";
77. this->label1->Size = System::Drawing::Size(26, 24);
78. this->label1->TabIndex = 2;
79. this->label1->Text = L"n:";
80. // button1
81. this->button1->Font = (gcnew
    System::Drawing::Font(L"Microsoft Sans Serif", 14.25F,
    System::Drawing::FontStyle::Regular,
    System::Drawing::GraphicsUnit::Point,
82. static_cast<System::Byte>(204)));
83. this->button1->Location = System::Drawing::Point(167,
    273);
84. this->button1->Name = L"button1";
85. this->button1->Size = System::Drawing::Size(195, 31);
86. this->button1->TabIndex = 3;
87. this->button1->Text = L"Hisoblash";
88. this->button1->UseVisualStyleBackColor = true;
89. this->button1->Click += gcnew
    System::EventHandler(this, &Form1::button1_Click);
90. //
91. // Form1
92. //
93. this->AutoScaleDimensions = System::Drawing::SizeF(6,
    13);
94. this->AutoScaleMode =
    System::Windows::Forms::AutoScaleMode::Font;
95. this->ClientSize = System::Drawing::Size(608, 330);
96. this->Controls->Add(this->button1);
97. this->Controls->Add(this->label1);
98. this->Controls->Add(this->textBox1);
99. this->Controls->Add(this->dataGridView1);
100. this->Name = L"Form1";
101. this->Text = L"DataGridView komponentasi bilan
    ishlash";
102. (cli::safe_cast<System::ComponentModel::ISupportIni
    tialize^ >(this->dataGridView1))->EndInit();
103. this->ResumeLayout(false);
104. this->PerformLayout();

105. }
106. #pragma endregion

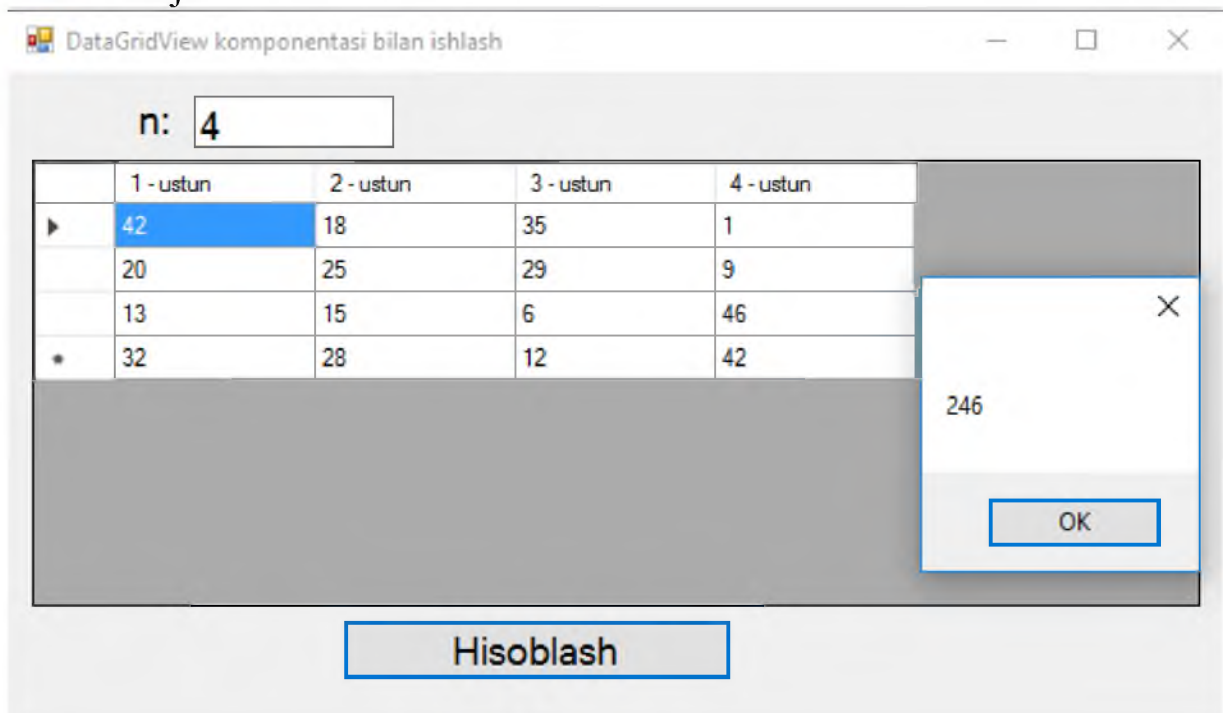
```

```

107. private: System::Void button1_Click(System::Object^
      sender, System::EventArgs^ e)
108. {
109.     int n=Convert::ToInt32(textBox1->Text);
110.     int s=0;
111.     dataGridView1->ColumnCount = n;
112.     dataGridView1->RowCount = n;
113.     int a[10][10];
114.     for(int i=0;i<n;i++){
115.         for(int j=0;j<n;j++){
116.             a[i][j]=rand()%50+1;
117.             if(a[i][j]%2==0)s+=a[i][j];
118.             dataGridView1->Columns[j]-
                >HeaderText=(j+1).ToString()+" - ustun";
119.             dataGridView1->Rows[i]->Cells[j]->Value
                =a[i][j].ToString();
120.         }
121.     }
122.     MessageBox::Show(Convert::ToString(s));
123. }
124. };
125. }

```

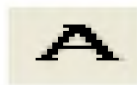
Dastur natijasi:



6.8- rasm. Dastur natijasi

20.2. Visual komponentalar yordamida amaliy dasturlar tuzish

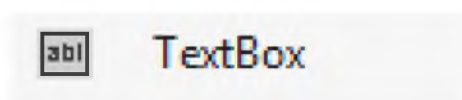
Label



Shaklda tahrir qilib bo'lmaydigan statik matnning to'rtburchak sohasini aks ettiradi. Odatda matn boshqa komponenta nomidan iborat bo'ladi.

Nom matni Text xususiyatining qiymatidir. **Alignment** xususiyati matnni tekislash usulini aniqlaydi. Shrift o'lchami avtomatik tarzda sohaning maksimal to'ldirilishiga mos kelishi uchun, **AutoSize** xususiyatining **true** qiymatini o'rnatish. Kalta soha ichida matnning hammasini ko'rish imkoniga ega bo'lish uchun, **WordWrap** xususiyatiga **true** qiymatini bering. **Transparent** xususiyatiga **true** qiymatini o'rnatish, boshqa komponentaning bir qismini to'g'ri uning ustida joylashtirilgan nom orasidan ko'rinib turadigan qilishingiz mumkin.

TextBox



Axborot yakka satrining tahrir qilinayotgan kiritishidagi to'rtburchak sohani shaklda aks ettiradi. Tahrir sohasining ichidagi boshlang'ich narsalarni Text xususiyatining qiymati bo'lgan satr aniqlaydi.

TextBox komponentasi **TtextBox** sinfining to'g'ridan-to'g'ri hosilasi bo'lib, uning barcha xususiyatlari, metodlari va voqealariga vorislik qiladi.

Button



Yozuvli to'rtburchak tugmani yaratadi. Tugmacha bosilganda, dasturda biror-bir hatti-harakat nomlanadi (initsiallashtiriladi).

Tugmachalar ko'proq dialogli darchalarda qo'llanadi. **Default** xususiyatining **true** qiymati tomonidan tanlab olingan yashirin tugmacha, dialog darchasida har gal enter klavishi bosilganda **OnClick** voqea qayta ishlatgichini ishga tushiradi.

Cancel xususiyatining **true** qiymati tanlab olgan bekor qilish tugmachasi, dialog darchasida har gal **Escape** klavishi bosilganda, **OnClick** voqea qayta ishlatgichini ishga tushiradi.

Topshiriq

Ma'ruza va amaliyot darslarida olingan ma'lumotlarga asoslanib, o'z variantingiz uchun quyidagilarni bajarinig:

- a) 10- jadvaldan o'z variantini ko'chirib oling;
- b) quyidagi namuna asosida o'z variantingizdagi masalalarni dasturini tuzing.

10- jadval

<u>№</u>	<u>Varianlar</u>
1	1- formada a[N][M] massiv elementlarini [N; M] oraliqdagi tasodifiy sonlar bilan to'ldiring. N va M ni TextBox komponentasi yordamida kiriting. Bu massivning har bir satr elementlarini o'sish tartibida 2- formaning label komponentasida tartiblangan holda chiqaring.
2	1- formada a[N][M] massiv elementlarini [N; M] oraliqdagi tasodifiy sonlar bilan to'ldiring. N va M ni TextBox komponentasi yordamida kiriting. Uning eng katta va eng kichik elementini toping va 2- formaning label komponentasida chiqaring. Agarda bunday elementlar ko'p bo'lsa, ularning sonini 3- formaning label komponentasida chiqaring.
3	1- formada a[N][M] massiv elementlarini [N; M] oraliqdagi tasodifiy sonlar bilan to'ldiring. N va M ni TextBox komponentasi yordamida kiriting. Uning eng katta elementining nomer(indeks)ini 2- formaning label komponentasida chiqaring va eng kichik elementining nomer(indeks)ini 3- formaning label komponentasida chiqaring.
4	1- formada a[N][M] massiv elementlarini [N; M] oraliqdagi tasodifiy sonlar bilan to'ldiring. N va M ni TextBox komponentasi yordamida kiriting. Uning eng katta elementini 2- formaning label komponentasida chiqaring va eng kichik elementini 3- formaning label komponentasida chiqaring. Eng katta va eng kichik elementlarining yig'indisini 4- formaning label komponentasida chiqaring.
5	1- formada a[N][M] massiv elementlarini [N; M] oraliqdagi tasodifiy sonlar bilan to'ldiring. N va M ni TextBox komponentasi yordamida kiriting. Bu massivni ustun elementlarini o'sish tartibida 2- formaning label komponentasida chiqaring.
6	1- formada a[N][M] massiv elementlarini [N; M] oraliqdagi tasodifiy sonlar bilan to'ldiring. N va M ni TextBox komponentasi yordamida kiriting. Bu

	massivni bosh diagonal elementlarini o‘shish tartibida 2- formaning label komponentasida chiqaring. Albatta mos ravishda satrlari ham o‘zgarishi lozim.
7	1- formada $a[N][M]$ massiv elementlarini $[N; M]$ oraliqdagi tasodifiy sonlar bilan to‘ldiring. N va M ni TextBox komponentasi yordamida kiriting. Elementini qiymati 200 dan kattalari sonini 2- formaning label komponentasida chiqaring va ularning o‘rta arifmetigini 3- formaning label komponentasida chiqaring.
8	1- formada $a[N][M]$ massiv elementlarini $[N; M]$ oraliqdagi tasodifiy sonlar bilan to‘ldiring. N va M ni TextBox komponentasi yordamida kiriting. Bu massivning bosh diagonal elementlarini kamayish tartibida 2- formaning label komponentasida chiqaring.
9	1- formada $a[N][M]$ massiv elementlarini $[N; M]$ oraliqdagi tasodifiy sonlar bilan to‘ldiring. N va M ni TextBox komponentasi yordamida kiriting. Elementining qiymatini bir vaqtda 5 ga va 3 ga qoldiqsiz bo‘linadiganlarining soni 2- formaning label komponentasida chiqaring va ularni yig‘indisini 3- formaning label komponentasida chiqaring.
10	1- formada $a[N][M]$ massiv elementlarini $[N; M]$ oraliqdagi tasodifiy sonlar bilan to‘ldiring. N va M ni TextBox komponentasi yordamida kiriting. Massivni chap diagonal elementlarini kamayish tartibida 2- formaning label komponentasida chiqaring.
11	1- formada $a[N][M]$ massiv elementlarini $[N; M]$ oraliqdagi tasodifiy sonlar bilan to‘ldiring. N va M ni TextBox komponentasi yordamida kiriting. Massiv elementlari tublari sonini 2- formaning label komponentasida chiqaring va ularning yig‘indisini 3- formaning label komponentasida chiqaring.
12	1- formada $a[N][M]$ massiv elementlarini $[N; M]$ oraliqdagi tasodifiy sonlar bilan to‘ldiring. N va M ni TextBox komponentasi yordamida kiriting. Massivning eng katta va eng kichik elementlari sonini 2- formaning label komponentasida chiqaring.
13	1- formada $a[N][M]$ massiv elementlarini $[N; M]$ oraliqdagi tasodifiy sonlar bilan to‘ldiring. N va M ni TextBox komponentasi yordamida kiriting. Massivning eng oxirgi manfiy elementining nomerini aniqlang va chiqaring.
14	1- formada $a[N][M]$ massiv elementlarini $[N; M]$ oraliqdagi tasodifiy sonlar bilan to‘ldiring. N va M ni TextBox komponentasi yordamida kiriting. Massivning eng oxirgi musbat elementining nomerini 2- formaning label komponentasida chiqaring.
15	1- formada $a[N][M]$ massiv elementlarini $[N; M]$ oraliqdagi tasodifiy sonlar bilan to‘ldiring. N va M ni TextBox komponentasi yordamida kiriting. Massivning eng oxirgi manfiy elementining nomerini 2- formaning label komponentasida chiqaring.
16	1- formada $a[N][M]$ massiv elementlarini $[N; M]$ oraliqdagi tasodifiy sonlar bilan to‘ldiring. N va M ni TextBox komponentasi yordamida kiriting. Massivning musbat elementlarining yig‘indisini 2- formaning label komponentasida chiqaring.

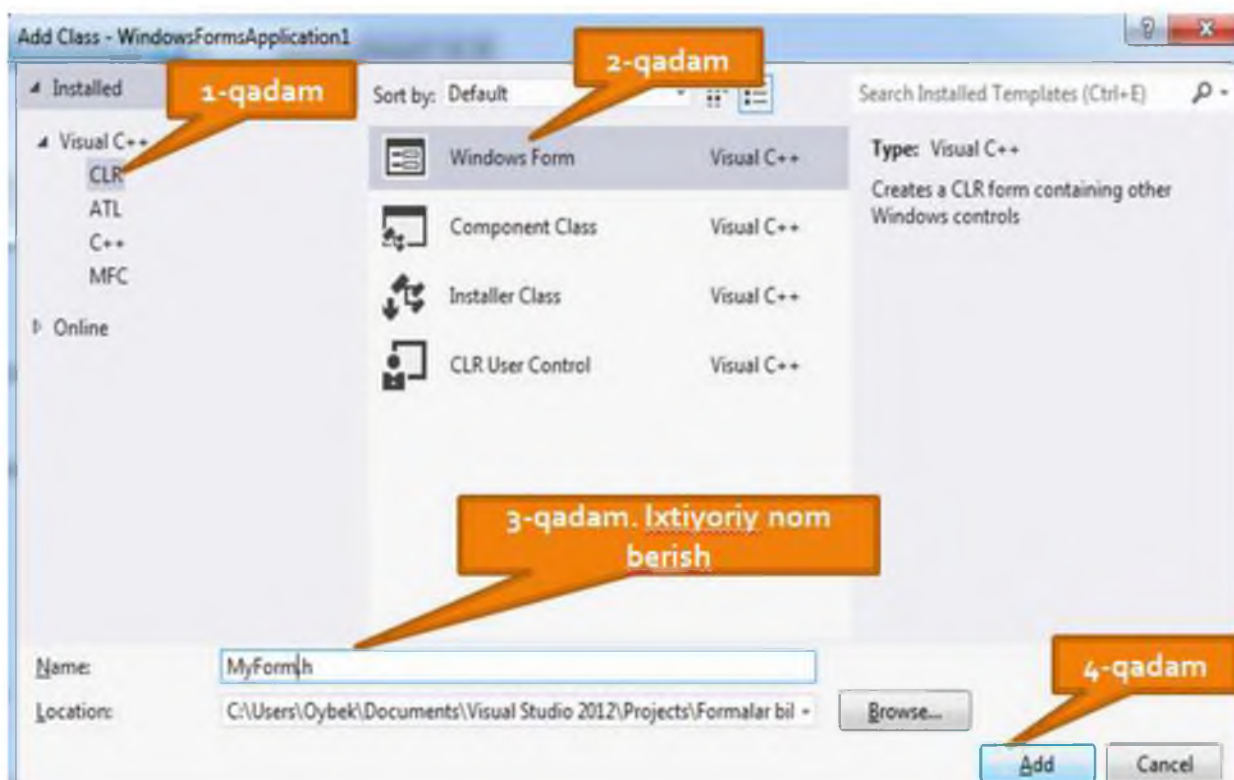
	komponentasida chiqaring va manfiy elementlarining yig'indisini 3-formaning label komponentasida chiqaring.
17	1- formada $a[N][M]$ massiv elementlarini $[N; M]$ oraliqdagi tasodifiy sonlar bilan to'ldiring. N va M ni TextBox komponentasi yordamida kiriting. Massivning eng oxirgi manfiy elementining nomerini 2- formaning label komponentasida chiqaring.
18	1- formada $a[N][M]$ massiv elementlarini $[N; M]$ oraliqdagi tasodifiy sonlar bilan to'ldiring. N va M ni TextBox komponentasi yordamida kiriting. Massivning eng oxirgi musbat elementining nomerini 2- formaning label komponentasida chiqaring.
19	1- formada $a[N][M]$ massiv elementlarini $[N; M]$ oraliqdagi tasodifiy sonlar bilan to'ldiring. N va M ni TextBox komponentasi yordamida kiriting. Berilgan massiv elementlarining a va b oraliqdagilari sonini 2- formaning label komponentasida chiqaring. a va b ni TextBox komponentasi yordamida kiriting.
20	1- formada $a[N][M]$ massiv elementlarini $[N; M]$ oraliqdagi tasodifiy sonlar bilan to'ldiring. N va M ni TextBox komponentasi yordamida kiriting. Massivni manfiy elementlarini ularning absolyut qiymati bilan almashtiring va almashtirishlar sonini 2- formaning label komponentasida chiqaring.
21	1- formada $a[N][M]$ massiv elementlarini $[N; M]$ oraliqdagi tasodifiy sonlar bilan to'ldiring. N va M ni TextBox komponentasi yordamida kiriting. Massivning manfiy elementlaridan yangi matritsani hosil qiling va 2-formaning dataGridView komponentasida chiqaring.
22	1- formada $a[N][M]$ massiv elementlarini $[N; M]$ oraliqdagi tasodifiy sonlar bilan to'ldiring. N va M ni TextBox komponentasi yordamida kiriting. Bu massiv elementlaridan 3 ga va 11 ga bo'linadiganlarining yig'indisi 2-formaning label komponentasida chiqaring.
23	$a[N][M]$ massiv elementlarini $[-1000; 1000]$ oraliqdagi tasodifiy sonlar bilan to'ldiring. Massiv elementlarining o'rta arifmetigini toping.
24	1- formada $a[N][M]$ massiv elementlarini $[N; M]$ oraliqdagi tasodifiy sonlar bilan to'ldiring. N va M ni TextBox komponentasi yordamida kiriting. Bu massiv elementlaridan 3 ga bo'linadiganlarining o'rta arifmetigini 2-formaning label komponentasida chiqaring va o'rta geometrigini 3-formaning label komponentasida chiqaring.
25	1- formada $a[N][M]$ massiv elementlarini $[N; M]$ oraliqdagi tasodifiy sonlar bilan to'ldiring. N va M ni TextBox komponentasi yordamida kiriting. Massivni elementlaridan musbatlarini alohida S massivga joylashtirib 2-formaning dataGridView komponentasida chiqaring va manfiylarini V massivga joylashtirib 3- formaning dataGridView komponentasida chiqaring.
26	1- formada $a[N][M]$ massiv elementlarini $[N; M]$ oraliqdagi tasodifiy sonlar bilan to'ldiring. N va M ni TextBox komponentasi yordamida kiriting.

	Massivni elementlarini teskari tartibda 2- formaning dataGridView komponentasida chiqaring.
27	1- formada $a[N][M]$ massiv elementlarini $[N; M]$ oraliqdagi tasodifiy sonlar bilan to'ldiring. N va M ni TextBox komponentasi yordamida kiriting. Massiv elementlarining birinchi musbatlarini 2- formaning dataGridView komponentasida chiqaring keyin manfiylarini 3- formaning dataGridView komponentasida chiqaring.
28	1- formada $a[N][M]$ massiv elementlarini $[N; M]$ oraliqdagi tasodifiy sonlar bilan to'ldiring. N va M ni TextBox komponentasi yordamida kiriting. Massivni eng katta va eng kichik elementlari orasidagi elementlarni 2- formaning dataGridView komponentasida chiqaring.
29	1- formada $a[N][M]$ massiv elementlarini $[N; M]$ oraliqdagi tasodifiy sonlar bilan to'ldiring. N va M ni TextBox komponentasi yordamida kiriting. Massivning eng katta va eng kichik elementlari orasida bo'lmagan elementlarni 2- formaning dataGridView komponentasida chiqaring.
30	1- formada $a[N][M]$ massiv elementlarini $[N; M]$ oraliqdagi tasodifiy sonlar bilan to'ldiring. N va M ni TextBox komponentasi yordamida kiriting. Massivning 7 ga bo'linadigan elementlarini massivdan o'chirib o'ringa 0 yozing va natijani 2- formaning dataGridView komponentasida chiqaring.
31	1- formada $a[N][M]$ massiv elementlarini $[N; M]$ oraliqdagi tasodifiy sonlar bilan to'ldiring. N va M ni TextBox komponentasi yordamida kiriting. Massiv elementlaridan qiymati toq bo'lganlarining yig'indisini 2- formaning label komponentasida chiqaring.
32	1-formada $a[N][M]$ massiv elementlarini $[N; M]$ oraliqdagi tasodifiy sonlar bilan to'ldiring. N va M ni TextBox komponentasi yordamida kiriting. Massiv elementlaridan qiymati juft bo'lganlarining yig'indisini 2-formaning label komponentasida chiqaring.

10- jadvaldagi topshiriqlarni bajarish uchun uslubiy qo'llanmalar.

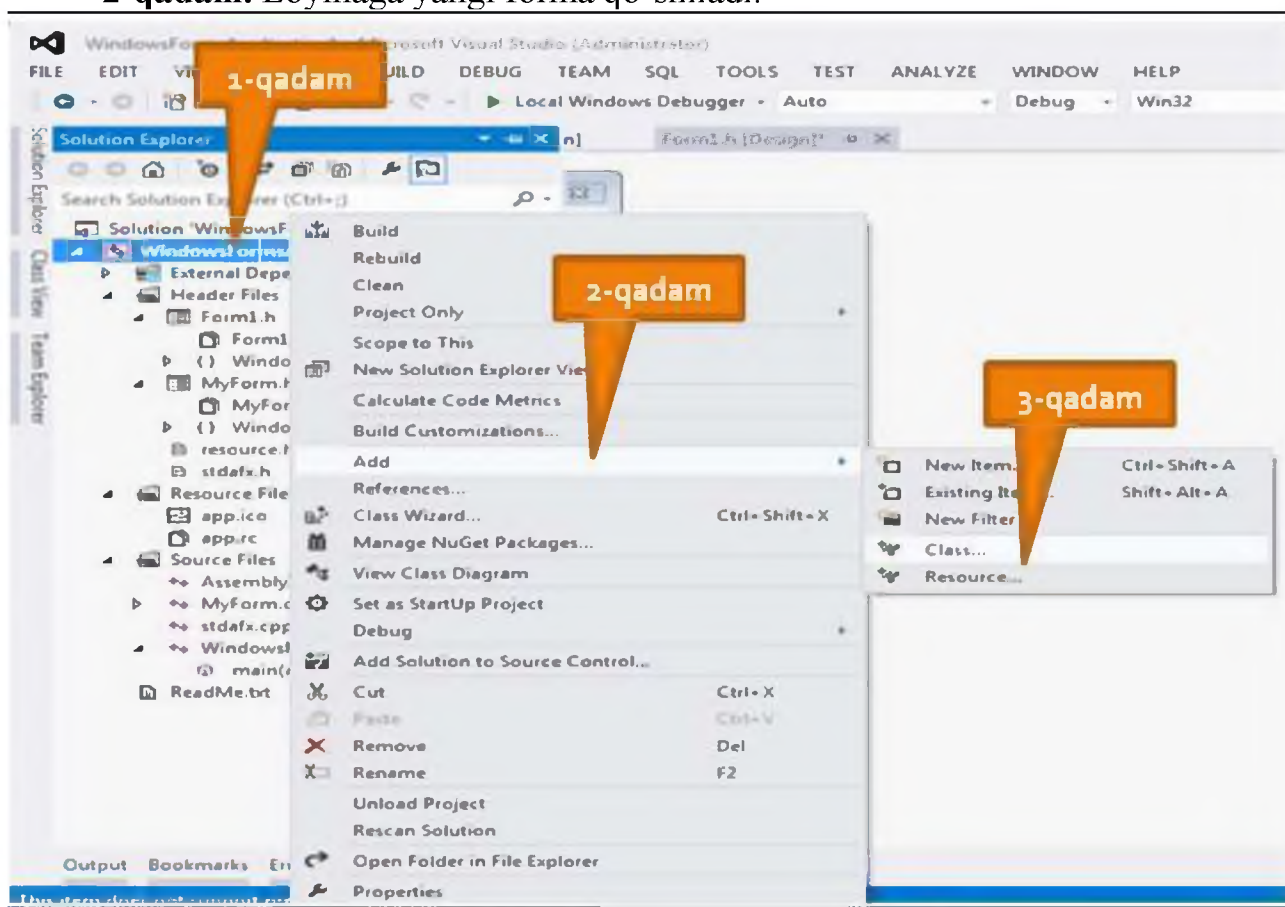
Visual C++ dasturida 32 – variantdagi masalani Windows Application muhitida ishlanishi:

1-qadam. Yangi forma ochiladi.



6.9 – rasm. Yangi forma ochish oynasi

2-qadam. Loyihaga yangi forma qo‘shiladi.



6.10 – rasm. Yangi forma qo‘shish oynasi

3-qadam. 1-formaga 2 ta **TextBox** komponentalari (misoldagi N va M uchun) joylashtiriladi:

4-qadam. 1-formaga 2 ta **dataGridView** komponentasi (misoldagi N va M o'lchovli massiv elementlarini chiqarish uchun) joylashtiriladi:

5-qadam. 1-formaga **button1** tugmachasi joylashtiriladi va **button1** tugmachasini 2 marta bosib, kodlar oynasida quyidagi kodlar teriladi:

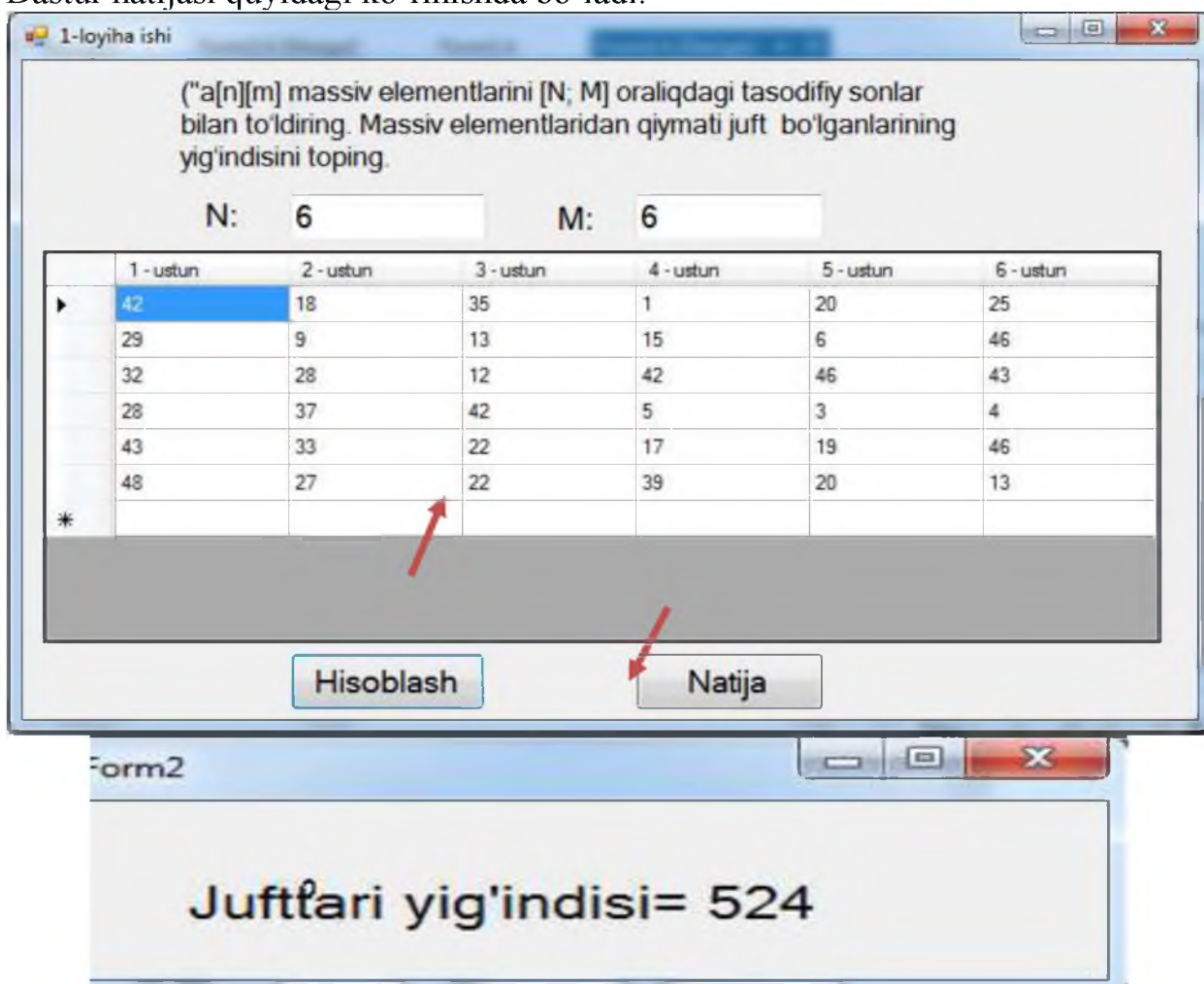
```
11. #include "iostream"
12. #include "conio.h"
13. #include "stdlib.h"
14. #include "form2.h"
15. ...
16. ...
17. int s;
18. #pragma endregion
19. private: System::Void button1_Click(System::Object^
    sender, System::EventArgs^ e) {
20. int a[10][10];
21. s=0;
22. int n=Convert::ToInt32(textBox1->Text);
23. int m=Convert::ToInt32(textBox2->Text);
24. dataGridView1->ColumnCount = n;
25. dataGridView1->RowCount = n;
26. for(int i=0;i<n;i++){
27. for(int j=0;j<m;j++){
28. a[i][j]=rand()%50+1;
29. dataGridView1->Columns[j]-
    >HeaderText=(j+1).ToString()+" - ustun";
30. dataGridView1->Rows[i]->Cells[j]->Value
    =a[i][j].ToString();
31. if(a[i][j]%2==0){s+=a[i][j];}
32. }}
33. }
```

6-qadam. 2- formaga **label** komponentasi joylashtiriladi va uning **Modifiers** xossasi **Public** ga o'zgartiriladi. Eslatma: Agar xossasi **Public** qilinmasa 1-formadan turib 2-forma komponentasini boshqarishga ruxsat berilmaydi.

7-qadam. 1- formaga **button2** tugmachasini joylashtiriladi va uni sichqonchanning chap tugmasi bilan 2 marta bosib, kodlar oynasida quyidagi kodlar teriladi:

1. `private: System::Void button2_Click(System::Object^ sender, System::EventArgs^ e) {`
2. `Form2^ open=gcnnew Form2();`
3. `this->Hide();`
4. `open->Show();`
5. `open->label1->Text="Juftlari yig'indisi= " + s.ToString();}`

Dastur natijasi quyidagi ko'rinishda bo'ladi:



6.11 – rasm. Dastur oynasi.


6.3. Visual C++ muhitining komponentalari: CheskBox, ChekedListBox, ComboBox komponentalarining xossalari (Properties) va hodisalari(Events)

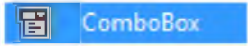
Ishning maksadi

CheskBox, CheckedListBox va **ComboBox** komponentalaridan foydalanib, amaliy vizual dastur yaratish.

CheckBox, CheckedListBox va **ComboBox** komponentalari bir nechta variantlardan bittasini yoki bir nechtasini tanlash yoʻli bilan ishlovchi vizual dasturlar yaratishi imkoniyatlarini beradi.

CheckBox komponentasi  koʻrinishida boʻladi. Uning asosiy xossalaridan biri Text va Checked lardir. Ushbu xossalari yordamida tanlanganligi aniqlanadi.

CheckedListBox komponentasi  koʻrinishida boʻladi. Uning asosiy xossalaridan biri Items dir. Ushbu xossasi yordamida tanlanganligi aniqlanadi.

ComboBox komponentasi  koʻrinishida boʻladi. Uning asosiy xossalaridan biri **Items** dir. Ushbu xossasi yordamida tanlanganligi aniqlanadi.

Ushbu komponentalarining tanlash parametrlari turlicha boʻlib, ularni kodi quyidagicha:

```
1. if(checkBox1->Checked && checkBox2->Checked){...}
2. void index(int indexChecked[]){
3. int i=0; IEnumerator^ myEnum1 =
4. checkedListBox1->CheckedIndices->GetEnumerator();
5. while( myEnum1->MoveNext())
6. {
7. indexChecked[i] = *safe_cast<Int32^>(myEnum1->
   Current); i++; }
8. }
9. index(indexChecked);
```

```

10.    if ( indexChecked[0]==0 && indexChecked[1]==1) {...}
11.if(comboBox1->Text=="Tasodifiy          sonlar          bilan
    to'ldirish"){...}

```

Topshiriq

Ma'ruza va amaliyot darslarida olingan ma'lumotlarga asoslanib, o'z variantingiz uchun quyidagilarni bajarinig:

- a) 15- jadvaldan o'z variantini ko'chirib oling;
- b) quyidagi namuna asosida o'z variantingizdagi masalalarni dasturini tuzing.

Har bir variant quyidagi ketma ketliklarda bajariladi:

- 1) **1- formada** 3 ta button komponentalari joylashtiriladi va 1- button ning text xossasi **CheckBox** ga, 2- button ning text xossasi **CheckedListBox** ga, 3- button ning text xossasi **ComboBox** ga o'zgartiriladi. 1- button ning **Click** hodisasiga `Form2^ open=gnew Form2(); this->Hide(); open->Show();`, 2- button ning **Click** hodisasiga `Form3^ open=gnew Form3(); this->Hide(); open->Show();`, 3- button ning **Click** hodisasiga `Form31^ open=gnew Form31(); this->Hide(); open->Show();` kodlari kiritiladi. Formalar bir biri bilan bog'lanadi.
- 2) **2- formada** 1 ta **dataGridView**, 2 ta **TextBox**, 1 ta **button** va 6 ta **checkBox** komponentalari joylashtiriladi. **checkBox1** ning Text xossasi "Hisoblash" ga, **checkBox2** ning Text xossasi "Tasodifiy sonlar bilan to'ldirish" ga, **checkBox3** ning Text xossasi "DadaGridView dan kiritish" ga, **checkBox4** ning Text xossasi "Sariq rang", **checkBox5** ning Text xossasi "Qizil rang" ga, **checkBox6** ning Text xossasi "Yashil rang" ga o'zgartiriladi va 15- jadvaldagi 3, 4 va 5- qadamlar bajariladi.
- 3) **3- formada** 1 ta **dataGridView**, 2 ta **TextBox**, 1 ta **button** va 1 ta **CheckedListBox** komponentalarini joylashtiring. **CheckedListBox** ning Itesm xossasiga "Hisoblash", "Tasodifiy sonlar bilan to'ldirish", "DadaGridView dan kiritish", "Sariq rang", "Qizil rang", "Yashil rang" iboralar kiritiladi va 15 -jadvaldagi 3, 4 va 5- qadamlar bajariladi.

- 4) **4- formada** 1 ta **dataGridView**, 2 ta **TextBox**, 1 ta **button**, 4 ta **checkBox**, 1 ta **comboBox** komponentalarini joylashtiring. **checkBox1** ning Text xossasi “Hisoblash” ga, **checkBox4** ning Text xossasi “Sariq rang”, **checkBox5** ning Text xossasi “Qizil rang” ga, **checkBox6** ning Text xossasi “Yashil rang” ga o‘zgartiriladi. **comboBox** ning **Items** xossasiga “Hisoblash”, “Sariq rang”, “Qizil rang”, “Yashil rang” iboralar kiritiladi va 11- jadvaldagi 3, 4 va 5- qadamlar bajariladi.

11- jadval

№	Varianlar
1	Yuqorida keltirilgan ketma ketlik asosida $a[N][M]$ massiv elementlarini $[N;M]$ oraliqdagi tasodifiy sonlar bilan va 2-, 3-, 4- formada dataGridView komponentasi yordamida to‘ldiring. N va M ni TextBox komponentasi yordamida kiriting. Bu massivning har bir satr elementlarini o‘shish tartibida 2-, 3-, 4- formada ma’lumotlar oynasi (<code>MessageBox::Show(“”)</code>)da tartiblangan holda chiqaring.
2	Yuqorida keltirilgan ketma ketlik asosida $a[N][M]$ massiv elementlarini $[N;M]$ oraliqdagi tasodifiy sonlar bilan va 2-, 3-, 4- formada dataGridView komponentasi yordamida to‘ldiring. N va M ni TextBox komponentasi yordamida kiriting. Uning eng katta va eng kichik elementini toping. Agarda bunday elementlar ko‘p bo‘lsa, ularning sonini 2-, 3-, 4- formada ma’lumotlar oynasi (<code>MessageBox::Show(“”)</code>) da tartiblangan holda chiqaring.
3	Yuqorida keltirilgan ketma ketlik asosida $a[N][M]$ massiv elementlarini $[N;M]$ oraliqdagi tasodifiy sonlar bilan va 2-, 3-, 4- formada dataGridView komponentasi yordamida to‘ldiring. N va M ni TextBox komponentasi yordamida kiriting. Uning eng katta elementining nomer(indeks)ini label komponentasida chiqaring va eng kichik elementining nomer(indeks)ini 2-, 3-, 4- formada ma’lumotlar oynasi (<code>MessageBox::Show(“”)</code>) da tartiblangan holda chiqaring.
4	Yuqorida keltirilgan ketma ketlik asosida $a[N][M]$ massiv elementlarini $[N;M]$ oraliqdagi tasodifiy sonlar bilan va 2-, 3-, 4- formada dataGridView komponentasi yordamida to‘ldiring. N va M ni TextBox komponentasi yordamida kiriting. Uning eng katta elementini, eng kichik elementini toping va eng katta va eng kichik elementlarining yig‘indisini 2-, 3-, 4- formada ma’lumotlar oynasi (<code>MessageBox::Show(“”)</code>)da tartiblangan holda chiqaring.
5	Yuqorida keltirilgan ketma ketlik asosida $a[N][M]$ massiv elementlarini $[N;M]$ oraliqdagi tasodifiy sonlar bilan va 2-, 3-, 4- formada dataGridView komponentasi yordamida to‘ldiring. N va M ni TextBox komponentasi

	yordamida kiriting. Bu massivni ustun elementlarini o'sish tartibida 2-, 3-, 4- formada ma'lumotlar oynasi (<code>MessageBox::Show("")</code>)da tartiblangan holda chiqaring.
6	Yuqorida keltirilgan ketma ketlik asosida $a[N][M]$ massiv elementlarini $[N; M]$ oraliqdagi tasodifiy sonlar bilan va 2-, 3-, 4- formada dataGridView komponentasi yordamida to'ldiring. N va M ni TextBox komponentasi yordamida kiriting. Bu massivni bosh diagonal elementlarini o'sish tartibida 2-, 3-, 4- formada ma'lumotlar oynasi (<code>MessageBox::Show("")</code>)da tartiblangan holda chiqaring. Albatta mos ravishda satrlari ham o'zgarishi lozim.
7	Yuqorida keltirilgan ketma ketlik asosida $a[N][M]$ massiv elementlarini $[N; M]$ oraliqdagi tasodifiy sonlar bilan va 2-, 3-, 4- formada dataGridView komponentasi yordamida to'ldiring. N va M ni TextBox komponentasi yordamida kiriting. Elementini qiymati 200 dan kattalari sonini va ularning o'rta arifmetigini 2-, 3-, 4- formada ma'lumotlar oynasi (<code>MessageBox::Show("")</code>)da tartiblangan holda chiqaring.
8	Yuqorida keltirilgan ketma ketlik asosida $a[N][M]$ massiv elementlarini $[N; M]$ oraliqdagi tasodifiy sonlar bilan va 2-, 3-, 4- formada dataGridView komponentasi yordamida to'ldiring. N va M ni TextBox komponentasi yordamida kiriting. Bu massivning bosh diagonal elementlarini kamayish tartibida 2-, 3-, 4- formada ma'lumotlar oynasi (<code>MessageBox::Show("")</code>)da tartiblangan holda chiqaring.
9	Yuqorida keltirilgan ketma ketlik asosida $a[N][M]$ massiv elementlarini $[N; M]$ oraliqdagi tasodifiy sonlar bilan va 2-, 3-, 4- formada dataGridView komponentasi yordamida to'ldiring. N va M ni TextBox komponentasi yordamida kiriting. Elementining qiymatini bir vaqtda 5 ga va 3 ga qoldiqsiz bo'linadiganlarining soni va ularni yig'indisini 2-, 3-, 4- formada ma'lumotlar oynasi (<code>MessageBox::Show("")</code>)da tartiblangan holda chiqaring.
10	Yuqorida keltirilgan ketma ketlik asosida $a[N][M]$ massiv elementlarini $[N; M]$ oraliqdagi tasodifiy sonlar bilan va 2-, 3-, 4- formada dataGridView komponentasi yordamida to'ldiring. N va M ni TextBox komponentasi yordamida kiriting. Massivni chap diagonal elementlarini kamayish tartibida 2-, 3-, 4- formada dataGridView da tartiblangan holda chiqaring.
11	Yuqorida keltirilgan ketma ketlik asosida $a[N][M]$ massiv elementlarini $[N; M]$ oraliqdagi tasodifiy sonlar bilan 2-, 3-, 4- formada dataGridView komponentasi yordamida to'ldiring. N va M ni TextBox komponentasi yordamida kiriting. Massiv elementlari tublari sonini va ularning yig'indisini 2-, 3-, 4- formada ma'lumotlar oynasi (<code>MessageBox::Show("")</code>)da tartiblangan holda chiqaring.
12	Yuqorida keltirilgan ketma ketlik asosida $a[N][M]$ massiv elementlarini $[N; M]$ oraliqdagi tasodifiy sonlar bilan 2-, 3-, 4- formada dataGridView komponentasi yordamida to'ldiring. N va M ni TextBox komponentasi

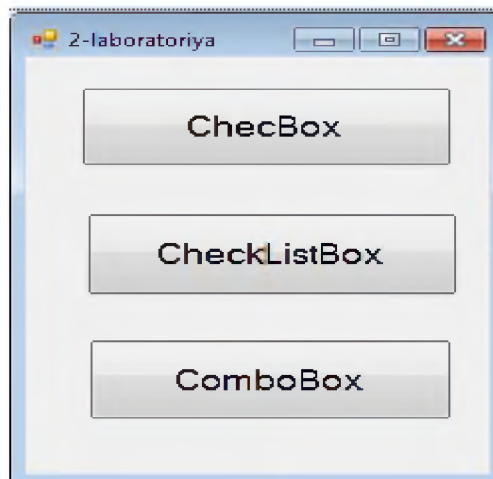
	yordamida kiriting. Massivning eng katta va eng kichik elementlari sonini 2-, 3-, 4- formada ma'lumotlar oynasi (<code>MessageBox::Show(" ")</code>)da tartiblangan holda chiqaring.
13	Yuqorida keltirilgan ketma ketlik asosida $a[N][M]$ massiv elementlarini $[N; M]$ oraliqdagi tasodifiy sonlar bilan 2-, 3-, 4- formada dataGridView komponentasi yordamida to'ldiring. N va M ni TextBox komponentasi yordamida kiriting. Massivning eng oxirgi manfiy elementining nomerini aniqlang va 2-, 3-, 4- formada ma'lumotlar oynasi (<code>MessageBox::Show(" ")</code>)da tartiblangan holda chiqaring.
14	Yuqorida keltirilgan ketma ketlik asosida $a[N][M]$ massiv elementlarini $[N; M]$ oraliqdagi tasodifiy sonlar bilan 2-, 3-, 4- formada dataGridView komponentasi yordamida to'ldiring. N va M ni TextBox komponentasi yordamida kiriting. Massivning eng oxirgi musbat elementining nomerini 2-, 3-, 4- formada ma'lumotlar oynasi (<code>MessageBox::Show(" ")</code>)da tartiblangan holda chiqaring.
15	Yuqorida keltirilgan ketma ketlik asosida $a[N][M]$ massiv elementlarini $[N; M]$ oraliqdagi tasodifiy sonlar bilan 2-, 3-, 4- formada dataGridView komponentasi yordamida to'ldiring. N va M ni TextBox komponentasi yordamida kiriting. Massivning eng oxirgi manfiy elementining nomerini 2-, 3-, 4- formada ma'lumotlar oynasi (<code>MessageBox::Show(" ")</code>)da tartiblangan holda chiqaring.
16	Yuqorida keltirilgan ketma ketlik asosida $a[N][M]$ massiv elementlarini $[N; M]$ oraliqdagi tasodifiy sonlar bilan 2-, 3-, 4- formada dataGridView komponentasi yordamida kiriting. N va M ni TextBox komponentasi yordamida kiriting. Massivning musbat elementlarining yig'indisini va manfiy elementlarining yig'indisini 2-, 3-, 4- formada ma'lumotlar oynasi (<code>MessageBox::Show(" ")</code>)da tartiblangan holda chiqaring.
17	Yuqorida keltirilgan ketma ketlik asosida $a[N][M]$ massiv elementlarini $[N; M]$ oraliqdagi tasodifiy sonlar bilan 2-, 3-, 4- formada dataGridView komponentasi yordamida to'ldiring. N va M ni TextBox komponentasi yordamida kiriting. Massivning eng oxirgi manfiy elementining nomerini 2-, 3-, 4- formada ma'lumotlar oynasi (<code>MessageBox::Show(" ")</code>)da tartiblangan holda chiqaring.
18	Yuqorida keltirilgan ketma ketlik asosida $a[N][M]$ massiv elementlarini $[N; M]$ oraliqdagi tasodifiy sonlar bilan 2-, 3-, 4- formada dataGridView komponentasi yordamida to'ldiring. N va M ni TextBox komponentasi yordamida kiriting. Massivning eng oxirgi musbat elementining nomerini 2-, 3-, 4- formada ma'lumotlar oynasi (<code>MessageBox::Show(" ")</code>)da tartiblangan holda chiqaring.
19	Yuqorida keltirilgan ketma ketlik asosida $a[N][M]$ massiv elementlarini $[N; M]$ oraliqdagi tasodifiy sonlar bilan 2-, 3-, 4- formada dataGridView komponentasi yordamida to'ldiring. N va M ni TextBox komponentasi yordamida kiriting. Berilgan massiv elementlarining a va b oraliqdagilari

	sonini 2-, 3-, 4- formada ma'lumotlar oynasi (<code>MessageBox::Show("")</code>)da tartiblangan holda chiqaring. a va b ni TextBox komponentasi yordamida kiriting.
20	Юқорида келтирилган кетма кетлик асосида $a[N][M]$ массив элементларини $[N; M]$ ораликдаги тасодифий сонлар билан 2-, 3-, 4- формада <code>dataGridView</code> компонентаси ёрдамида тўлдиринг. N ва M ни <code>TextBox</code> компонентаси ёрдамида киритинг. Массив элементларидан қиймати жуфт бўлганларининг йигиндисини 2, 3, 4- формада (<code>MessageBox::Show("")</code>)да тартибланган ҳолда чиқаринг.

11- jadvaldagi topshiriqlarni bajarish uchun uslubiy qo'llanmalar.

Visual C++ dasturida 20 – variantdagi masalani Windows Application muhitida ishlanishi:

1- qadam: 1- formaga boshqa formalarni bog'lash uchun quyidagi kodlar kiritiladi: `#include "Form2.h"; #include "Form3.h"; #include "Form31.h";`



4.12-rasm. 1- formaning ko'rinishi

1- forma 2- formaga bog'lanishi uchun **ChecBox** tugmasining **OnClick** hodisasiga quyidagi kodlar kiritiladi:

```
Form2^ open=gcnw Form2();
this->Hide();
open->Show();
```

1- forma 3- formaga bog'lanishi uchun **CheckListBox** tugmasining **OnClick** hodisasiga quyidagi kodlar kiritiladi:

```
Form3^ open=gcnw Form3();
```

```
this->Hide();
open->Show();
```

1- forma 3- formaga bog'lanish uchun **ComboBox** tugmasining **OnClick** hodisasiga quyidagi kodlar kiritiladi:

```
Form3^ open=gcnnew Form3();
this->Hide();
open->Show();
```

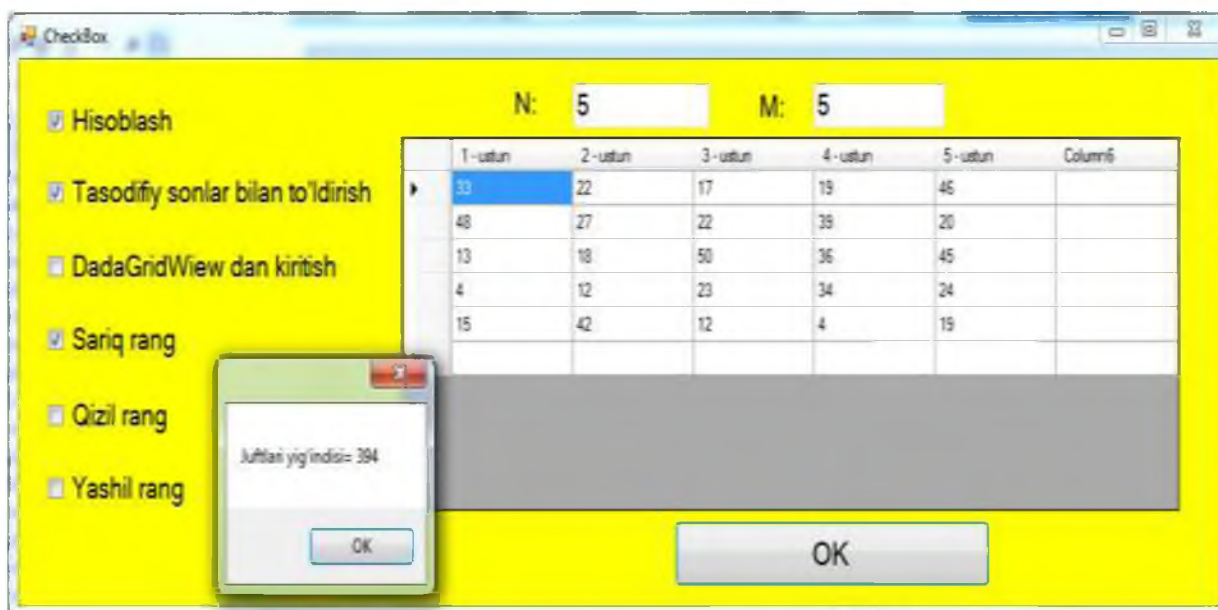
2- qadam: 32 – variantdagi masalani yechish uchun 2- formada **button** ning **OnClick** hodisasiga quyidagi kodlar kiritiladi:

```
1. int a[10][10];
2. s=0; int n,m;
3. if(textBox1->Text!="" && textBox2->Text!=""){
4. n=Convert::ToInt32(textBox1->Text);
5. m=Convert::ToInt32(textBox2->Text);}
6. if(checkBox1->Checked && checkBox2->Checked ){
7. if(checkBox4->Checked){Form2::BackColor=
  System::Drawing::Color::Yellow; }
8. if(checkBox5->Checked){Form2::BackColor=
  System::Drawing::Color::Red; }
9. if(checkBox6->Checked){Form2::BackColor=
  System::Drawing::Color::Green; }
10. for(int i=0;i<n;i++){
11. if(g==0){dataGridView1->Rows->Add();}
12. for(int j=0;j<m;j++){a[i][j]=rand()%50+1;
13. dataGridView1->Columns[j]->HeaderText=
  (j+1).ToString()+" - ustun";
14. dataGridView1->Rows[i]->Cells[j]->Value
  =a[i][j].ToString();
15. if(a[i][j]%2==0){s+=a[i][j];}
16. }} g+=1;
17. MessageBox::Show("Juftlari yig'indisi= " +
  s.ToString()); }
18. if(checkBox1->Checked && checkBox3->Checked){s=0;
  for(int i=0;i<n;i++){
19. if(g==0){dataGridView1->Rows->Add();}
20. for(int j=0;j<m;j++){
21. a[i][j]=Convert::ToInt32(dataGridView1->Rows[i]-
  >Cells[j]->Value );
22. if(a[i][j]%2==0){s+=a[i][j];} } }
```

```

23. MessageBox::Show("Juftlari yig'indisi= " +
    s.ToString());
24. }

```



4.12-rasm. 2- formaning ko‘rinishi

3- qadam: 32 – variantdagi masalani yechish uchun 3- formada button1 ning Click hodisasiga quyidagi kodlar kiritiladi:

```

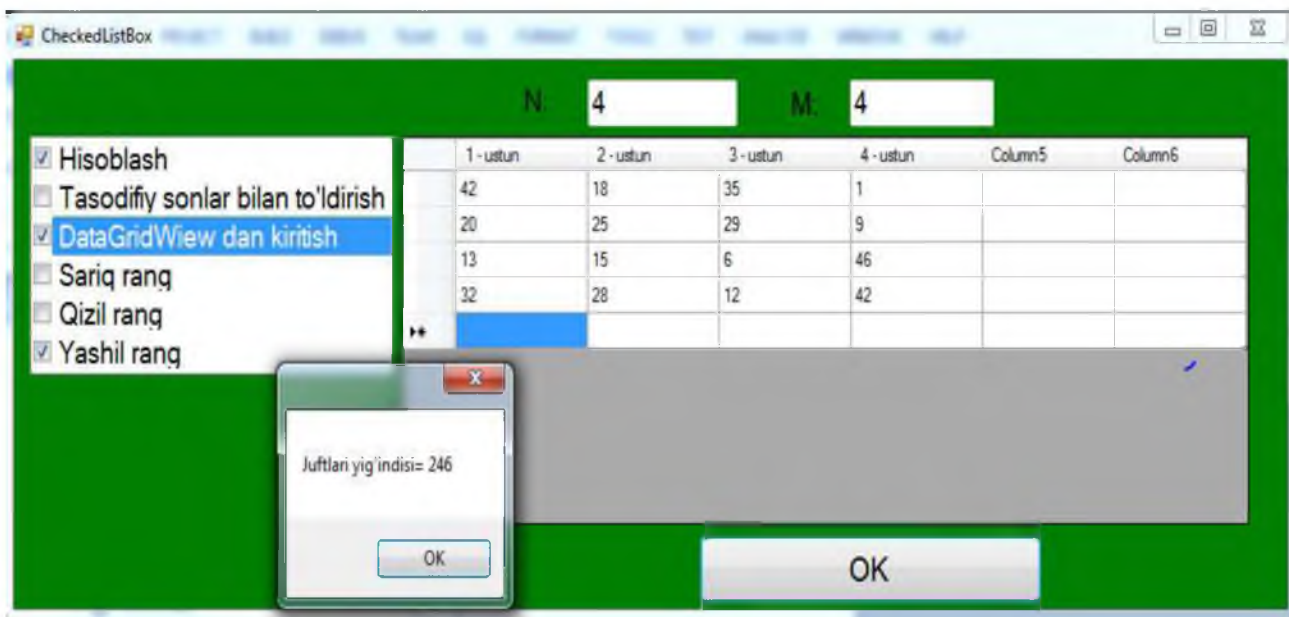
1. int g,s;
2. #pragma endregion
3. void index(int indexChecked[]){ int i=0;
4. IEnumerator^ myEnum1 = checkedListBox1>CheckedIndices-
    >GetEnumerator();
5. while( myEnum1->MoveNext()){
6. indexChecked[i] = *safe_cast<Int32^>(myEnum1->Current);
7. i++; } }
8. private: System::Void button1_Click(System::Object^
    sender, System::EventArgs^ e) {
9. int a[10][10]; int indexChecked[10];
10. s=0; int n,m;
11. if(textBox1->Text!="" && textBox2->Text!=""){
12. n=Convert::ToInt32(textBox1->Text);
13. m=Convert::ToInt32(textBox2->Text);}
    index(indexChecked);
14. if (indexChecked[0]==0 && indexChecked[1]==1){

```

```

15. if (indexChecked[2]==3{
    Form3::BackColor=System::Drawing::Color::Yellow; }
16. if (indexChecked[2]==4){
    Form3::BackColor=System::Drawing::Color::Red; }
17. if (indexChecked[2]==5){
    Form3::BackColor=System::Drawing::Color::Green; }
18. for(int i=0;i<n;i++){
19. if(g==0){dataGridView1->Rows->Add();}
20. for(int j=0;j<m;j++){
21. a[i][j]=rand()%50+1;
22. dataGridView1->Columns[j]->HeaderText=
    (j+1).ToString()+" - ustun";
23. dataGridView1->Rows[i]->Cells[j]->Value
    =a[i][j].ToString();
24. if(a[i][j]%2==0){s+=a[i][j];} }}
25. g+=1;
26. MessageBox::Show("Juftlari yig'indisi= " +
    s.ToString()); }
27. if ( indexChecked[0]==0 && indexChecked[1]==2){
28. if ( indexChecked[2]==3){
    Form3::BackColor=System::Drawing::Color::Yellow; }
29. if ( indexChecked[2]==4){
    Form3::BackColor=System::Drawing::Color::Red; }
30. if ( indexChecked[2]==5){
    Form3::BackColor=System::Drawing::Color::Green; }
31. s=0;
32. for(int i=0;i<n;i++){
33. if(g==0){dataGridView1->Rows->Add();}
34. for(int j=0;j<m;j++){
35. a[i][j]=Convert::ToInt32( dataGridView1->Rows[i]-
    >Cells[j]->Value );
36. if(a[i][j]%2==0){s+=a[i][j];} }}
37. MessageBox::Show("Juftlari yig'indisi= " +
    s.ToString()); }}

```



4.13-rasm. 3- formaning ko‘rinishi

4-qadam: 32 – variantdagi masalani yechish uchun 4-formada button ning Click hodisasiga quyidagi kodlar kiritiladi:

```

1. int g,s;
2. #pragma endregion
3. private: System::Void Form31_FormClosed(System::Object^
   sender, System::Windows::Forms::FormClosedEventArgs^ e)
   {
4. Application::Restart();
5. }
6. private: System::Void button1_Click(System::Object^
   sender, System::EventArgs^ e)
7. {
8. int a[10][10];
9. int indexChecked[10]; s=0; int n,m;
10. if(textBox1->Text!="" && textBox2->Text!="")
11. {
12. n=Convert::ToInt32(textBox1->Text);
13. m=Convert::ToInt32(textBox2->Text);}
14. if(comboBox1->Text=="Tasodifiy sonlar bilan
   to'ldirish")
15. {
16. if(checkBox1->Checked)
17. {
18. if(checkBox4->Checked)

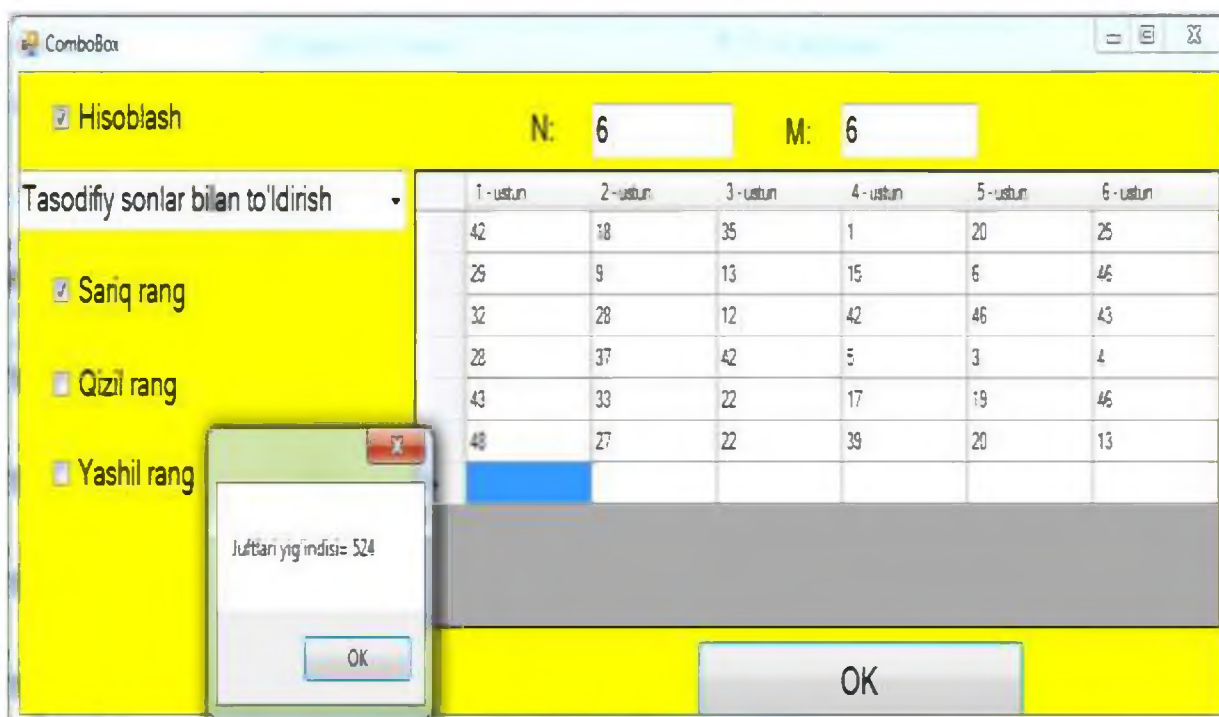
```

```

19. { Form31::BackColor=System::Drawing::Color::Yellow;
20. }
21. if(checkBox5->Checked)
22. { Form31::BackColor=System::Drawing::Color::Red;
23. }
24. if(checkBox6->Checked)
25. { Form31::BackColor=System::Drawing::Color::Green;
26. }
27. for(int i=0;i<n;i++){ if(g==0)
28. {
29. dataGridView1->Rows->Add();
30. }
31. for(int j=0;j<m;j++)
32. {
33. a[i][j]=rand()%50+1;
34. dataGridView1->Columns[j]->HeaderText=
    (j+1).ToString()+" - ustun";
35. dataGridView1->Rows[i]->Cells[j]->Value
    =a[i][j].ToString();
36. if(a[i][j]%2==0)
37. { s+=a[i][j];}} g+=1;
38. MessageBox::Show("Juftlari yig'indisi= " +
    s.ToString());}
39. }
40. if(comboBox1->Text=="DataGridWiew dan kiritish"){
41. if(checkBox1->Checked)
42. {
43. if(checkBox4->Checked)
44. { Form31::BackColor=System::Drawing::Color::Yellow;
45. }
46. if(checkBox5->Checked){
    Form31::BackColor=System::Drawing::Color::Red; }
47. if(checkBox6->Checked){
    Form31::BackColor=System::Drawing::Color::Green;}
48. s=0;
49. for(int i=0;i<n;i++){
50. if(g==0){dataGridView1->Rows->Add();
51. }
52. for(int j=0;j<m;j++){
53. a[i][j]=Convert::ToInt32( dataGridView1->Rows[i]-
    >Cells[j]->Value );
54. if(a[i][j]%2==0){s+=a[i][j];} }}

```

```
55. MessageBox::Show("Juftlari yig'indisi= " +
    s.ToString());}    }
```



4.14-rasm. 4-formaning ko‘rinishi

6.4. Visual C++ muhitida muloqot oynalari bilan ishlash: DateTimePiker, LinkLabel va ListBox komponentalarining xossalari (Properties) va hodisalari(Events)

Ishning maksadi

DateTimePiker, LinkLabel va ListBox komponentalaridan foydalanib, amaliy vizual dastur yaratish.

ListBox komponentasi:



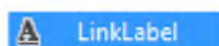
Tanlash, qo‘shish yoki o‘chirish uchun mo‘ljallangan matn variantlari ro‘yxatining to‘rtburchak sohasini aks ettiradi.

Agar ro'yxatdagi barcha elementlar ajratilgan sohaga sig'masa, ro'yxatni aylantirish lineykasi yordamida ko'rib chiqish mumkin. Ro'yxat elementlari **Items** xususiyatining ichida, dastur bajarilish vaqtida tanlab olinadigan element raqami esa **ItemIndex** xususiyatining ichida joylashgan bo'ladi. Ro'yxat elementlari matn muharririning darchasi **Items** xususiyati qiymatining grafasida tugmacha bilan ochiladi. Ro'yxat elementlarini **Items** ob'ektining **Add**, **Append**, **Delete** va **Insert** metodlari yordamida dinamik tarzda qo'shish, o'chirish, orasiga joylash va o'rnini almashtirish mumkin. Masalan:

ListBox1->Items->Add ("Ro'yxatning oxirgi elementi");

Sorted xususiyatining **true** qiymati ro'yxat elementlarini alifbo tartibida ajratib joylashtiradi.

LinkLabel komponentasi:



Ushbu komponenta yordamida turli hil havolalarga murojat qilish mumkin. Uning asosiy xossalardan biri **Items** va **LinkClicked** xossalari. Ushbu loyihada **Button** vazifasini bajaruvchi bir nechta imkoniyatlarni onclick hodisasiga yuklatiladi.

DateTimePicker komponentasi:

Ushbu komponenta tizim kalendaridan foydalanish imkoniyatini beradi. Uning turli parametrlari bilan tanlangan sanani o'zlashtirish imkoniyatlari bor. Ushbu imkoniyatlar **FormatChange** xossasida joylashgan. Misol: `dateTimePicker1->Format = DateTimePickerFormat::Custom; dateTimePicker1->CustomFormat = "dd.dd.yyyy";`

Topshiriq

Ma'ruza va amaliyot darslarida olingan ma'lumotlarga asoslanib, o'z variantingiz uchun quyidagilarni bajarinig:

- a) 16- jadvaldan o'z variantini ko'chirib oling;
- b) quyidagi namuna asosida o'z variantingizdagi masalalarni dasturini tuzing.

Har - bir variantda quyidagi ketma ketliklar bajariladi:

Har- bir variantda keltirilgan Kiruvchi, chiquvchi ma'lumotlar va qidirish parametrlarini dasturning asosiy oynasi(**Form1**)dan matnli faylga kirituvchi va ushbu ma'lumotlar asosida qidiruv parametrlarini tashkil qiladigan vizual dastur tuzing. Har - bir kiritiluvchi ma'lumotlarga mos alohida komponentalar(**TextBox**, **ComboBox**, **DateTimePiker**)dan mosini tanlab, formaga joylashtiring. **CheckedBox** va **ListBox** koponentalari yordamida qidirish parametrlarini o'rnatish.

12- jadval

№	Topshiriq nomi	Kiruvchi, chiquvchi ma'lumotlar va qidirish parametrlari
1.	Inson	Familiyasi, ismi, jinsi, millati, e'tiqod qiladigan dini, bo'yi vazni, tug'ilgan sanasi (yil, oy, kun), telefon raqami, uy manzili (pochta indeksi, mamlakat, viloyat, tuman, shahar, ko'cha, uy, xonadon).
2.	Maktab o'quvchisi	Familiyasi, ismi, otasining ismi, sinf nomeri, jinsi, tug'ilgan sanasi (yil, oy, kun), uy manzili, sinfi raxbari, bahosi.
3.	Xaridor	Familiyasi, ismi, otasining ismi, manzili, tuman, shahar, ko'cha, uy raqami, xonadon raqami, kredit kartochkasi raqami yoki hisob raqami.
4.	Bemor	Familiyasi, ismi, tug'ilgan yili, telefon raqami, tumani, uy manzili, tibbiy kartochka raqami, qon guruhi.
5.	Sport komandasi	Nomi, shahri, nechta o'yin o'ynaganligi, qancha ochko to'plaganligi (yutqazgan, yutgan, durang), o'yinchilar soni.
6.	Stadion	Nomi, sport turi, qurilgan sanasi, manzili, tomoshabinlar sig'imi, arenalar, maydonchalar soni, sektor raqami.
7.	Avtomobil egasi	Familiyasi, ismi, avtomobil raqami, avtomobil nomi, texpasport raqami, tug'ilgan yili, telefon raqami, qayd qilingan YPX bo'limi.
8.	Avtomobil	Markasi, rangi, seriyasi nomeri, qayd raqami, eshiklari soni, chiqqan yili, narxi.
9.	Film	Nomi, rejissori (familiyasi, ismi), mamlakati, ekranga chiqqan yili, narxi (ketgan xarajatlar), olingan daromad.
10.	Musiqiy mahsulot	Tashuvchi (gramplastinka, audiokasseta, lazerli disk), katalogdagi tartib raqami, nomi, ijrochisi (familiyasi, ismi), ijro davomiyligi, asarlar soni, katalog bo'yicha narxi.
11.	Vokal-cholg'u guruhi albomi	Guruhning nomi, albomning nomi, kasseta, diskdagi qo'shiqlar soni, albom chiqqan yil, ishlab chiqaruvchi firma.

12.	Davlat	Mamlakat nomi, poytaxti, davlat tili, aholisi (soni), yer maydoni, pul birligi valyutasining so‘m (dollar)ga nisbatan kursi, davlat tuzumi, qit‘asi.
13.	Monitor	Davlat, ishlab chiqarilgan yili, markasi, narxi, kafolat vaqti, o‘lchami, rangi, netto va boshqalar.
14.	Kitob	Muallifi, ham muallif, asar nomi, beti, yili, sohasi, janri, narxi va boshqalar.
15.	Antivirus	Nomi, versiyasi, narxi, kalit vaqti, firmasi va boshalar.
16.	Telefon	Nomi, markasi, yili, o‘lchami, narxi, kamera o‘lchami, operasion tizim tipi, garantiya, davlati va boshqalar.
17.	Qaydnoma	1-joriy nazorat (jn), 1-joriy nazorat qayta ishlash (jnqi), 1-oraliq nazorat (on), 1-oraliq nqi, 2-on, 2-onqi, jamon OT + ON, yakuniy nazorat, o‘zlashtirish ko‘rsatkichi, reyting bali.
18.	O‘yinchoq	Nomi, narxi, davlati, bolalar yoshiga chegara va boshqalar.
19.	Talaba	Familiyasi, ismi, sharifi, tug‘ilgan sanasi (kun oy yil), guruhi, bahosi va boshqalar.
20.	Yugurish	Familiya, ismi, sharifi, davlati, yoshi, masofa, shaxsiy rekordi vaqti vaqti, o‘rni.
21.	Printer	Nomi, markasi, davlati, ishlab chiqarilgan yili, 1 minutda chiqaradigan varaqlar soni, o‘g‘irligi, kartrij raqami, toneri, rangi, og‘irligi va boshqalar.
22.	RAM xotira	Nomi, o‘lchami, ishlab chiqarilgan yili, narxi, davlati va boshqalar.
23.	Futbolchi	Familiyasi, ismi, sharifi, tug‘ilgan sanasi (kun oy yil), fuqaroligi, ampulasi, klubi, gollar soni va boshqalar.
24.	Qog‘oz	Markasi, o‘lchami, narxi, og‘irligi, rangi, davlati va boshqalar.
25.	Shkaf	Nomi, turi, rangi, material, o‘lchami, narxi, davlati va boshqalar.
26.	Daftar	Nomi, varaqlar soni, qog‘ozi, firmasi, abloshkasi (qattiq, silliq), rangi, o‘lchami va boshqalar
27.	Univer-sitet	Nomi, tashkil topgan yili, manzili (viloyati, tumani, ko‘chasi, raqami), telefoni, faksi, talabalar soni va boshqalar.
28.	Bino	Nomi, manzili, qurilgan yili, qurgan tashkilot, ajratilgan mablag‘, yer maydoni, bino maydoni, etajlar soni, qurish usuli va boshqalar.
29.	Mahalla	Nomi, tumani, shirkati, aholi soni, oqsoqoli, telefoni, ko‘p qavatli uylar soni, oilalar soni va boshqalar.
30.	Mashina	Nomi, dvigatel nomi, sig‘imi, ot kuchi, rangi, narxi, davrlari va boshqalar.

12- jadvaldagi topshiriqlarni bajarish uchun uslubiy qo‘llanmalar

Visual C++ dasturida 19 – variantdagi masalani Windows Application muhitida ishlanishi:

1- qadam: Asosiy oyna formasida kiruvchi ma'lumotlarni saqlash uchun 1 ta **TextBox** komponentasini joylashtiring va uning **text** xossasiga C:\2.txt ko'zni kriting. Talabanning familiyasi, ismi, sharifini kiritish uchun 3 ta **TextBox** komponentasini, tug'ilgan sanasi(kun, oy, yil)ni kiritish uchun 1ta **DateTimePicker** komponentasini, guruhi va bahosini kiritish uchun 2 ta **ComboBox** komponentasini joylashtiring. **ComboBox1** ning **Items** xossasiga (214-16, 215-16, 216-17, 217-16, 218-16, 219-16, 220-16) guruh nomerlarini, **ComboBox2** ning **Items** xossasiga(3, 4, 5) baholarni kriting.

2- qadam: 7 ta **Label** va 1 ta **Button** komponentasi joylashtiriladi va sichqonchaning o'ng tugmasi ikki marta tez bosilib, kodlar oynasiga quyidagi kodlar yoziladi:

1. `textBox3->AppendText(j.ToString()+" - talaba"+Environment.NewLine);`
2. `textBox3->AppendText(textBox2->Text+Environment.NewLine);`
3. `textBox3->AppendText(textBox4->Text+Environment.NewLine);`
4. `textBox3->AppendText(textBox5->Text+Environment.NewLine);`
5. `textBox3->AppendText(comboBox1->Text+Environment.NewLine);`
6. `textBox3->AppendText(String.Format("{0}", dateTimePicker1->Text)+Environment.NewLine);`
7. `try{ textBox3->AppendText(comboBox2->Text+Environment.NewLine); }`
8. `catch(Exception^ vaziyat){ MessageBox::Show("Bahosini notogri kiritdingiz"); } j++;`

Uning tashqi ko'rinishi 4.1- rasmda keltirilgan:

6.15- rasm. Asossiy oynaning bir qismi

3- qadam: Ma'lumotlarni faylga yozish va fayldan ma'lumotlarni o'qish uchun 1 ta **TextBox** komponentasi joylashtiriladi va uning **MultiLine** xossasi **true** ga o'zgartiriladi. 2 ta **Button** komponentasi joylashtiriladi, 1- sini ustida sichqonchaning o'ng tugmasi ikki marta tez bosilib quyidagi kodlar yoziladi:

```


1. filename = textBox1->Text;
2. try{
3. // Faylga yozish uchun obyekt yaratish
4. System::Text::Encoding^ shirift
   =System::Text::Encoding::GetEncoding(1251);
5. IO::StreamReader^ oquvchi = gcnew
   IO::StreamReader(filename, shirift);
6. textBox3->Text = oquvchi->ReadToEnd(); oquvchi->Close();
   }
7. catch (IO::FileNotFoundException^ vaziyat){
   MessageBox::Show(vaziyat->Message + "\nBunday fayl
   yo'q", "Xatolik", MessageBoxButtons::OK,
   MessageBoxIcon::Exclamation); }
8. catch (Exception^ vaziyat){ MessageBox::Show(vaziyat-
   >Message, "Xatolik",
9. MessageBoxButtons::OK, MessageBoxIcon::Exclamation);}

```

10.2- sini ustida sichqonchanning o'ng tugmasi ikki marta tez bosilib quyidagi kodlar yoziladi:

```
11. filename = textBox1->Text;
12. try{ auto shirift
    =System::Text::Encoding::GetEncoding(1251);
13. auto yozuvch = gcnew IO::StreamWriter(filename,
    false,shirift);
14. yozuvch->Write(textBox3->Text); yozuvch->Close(); }
15. catch (Exception^ vaziyat){ MessageBox::Show(vaziyat-
    >Message, "Xatolik",
16. MessageBoxButtons::OK, MessageBoxIcon::Exclamation); }
```

1- sining **text** xossasiga **Fayldan o'qish** matnini, 2- sining **text** xossasiga **Faylga yozish** matnini kiriting. 3-qadamdan keyin asosiy oyna ko'rinishi 4.16-rasmdagidek bo'ladi:



6.16- rasm. Asosiy oynaning 2 ta qismini ko'rinishi

4- qadam: Qidirish parametrlarini o'rnatish uchun 1 ta **Label** komponentasi joylashtiriladi va uning **text** xossasiga **Qidirish turini tanlang** matnini kiritiladi. 6 ta **CheckBox**, 1 ta **TextBox**, 1 ta **Button** komponentasi joylashtiriladi va sichqoncha o'ng tugmasi ikki marta tez bosilib, quyidagi kodlar yoziladi:

```

1. static int j=1; int y,qq; bool e;
2. void qidirish(char *FileName){ e=true;
3. Form1::textBox3->Clear();
4. if(textBox6->Text!=""){
5. FILE *oqi; oqi=fopen(FileName,"r");
6. if(!oqi){ MessageBox::Show("Xato"); return;}
7. char temp[40];
8. char ism[40];
9. char fam[40];
10. char sharfi[40];
11. char guruh[40];
12. char Tu_sana[40];
13. int baho;
14. String^ Tt,^Ti,^Tf,^Ts,^Tg,^Tu,^Tb;
15. while(!feof(oqi)){
16. int n;
17. fgets(temp,25,oqi);
18. fgets(fam,25,oqi);
19. fgets(ism,25,oqi);
20. fgets(sharfi ,25,oqi);
21. fgets(guruh ,25,oqi);
22. fgets(Tu_sana,25,oqi);
23. fscanf(oqi,"%d",&baho);
24. if(Form1::checkBox1->Checked==true){y=1;
25. char *satr=and_SysStringToChar(textBox6->Text);
26. if(strstr(fam,satr)!=NULL){qq=1;}
27. else {qq=0;}}
28. else if(Form1::checkBox2->Checked==true){y=2;
29. char *satr=and_SysStringToChar(textBox6->Text);
30. if(strstr(ism,satr)!=NULL){qq=1;}
31. else {qq=0;}}
32. else if(Form1::checkBox3->Checked==true){y=3;
33. char *satr=and_SysStringToChar(textBox6->Text);
34. if(strstr(sharfi,satr)!=NULL){qq=1;}
35. else {qq=0;}}
36. else if(Form1::checkBox4->Checked==true){y=4;
37. char *satr=and_SysStringToChar(textBox6->Text);
38. if(strstr(guruh,satr)!=NULL){qq=1;}
39. else {qq=0;}}
40. else if(Form1::checkBox5->Checked==true){y=5;
41. char *satr=and_SysStringToChar(textBox6->Text);
42. if(strstr(Tu_sana,satr)!=NULL){qq=1;}

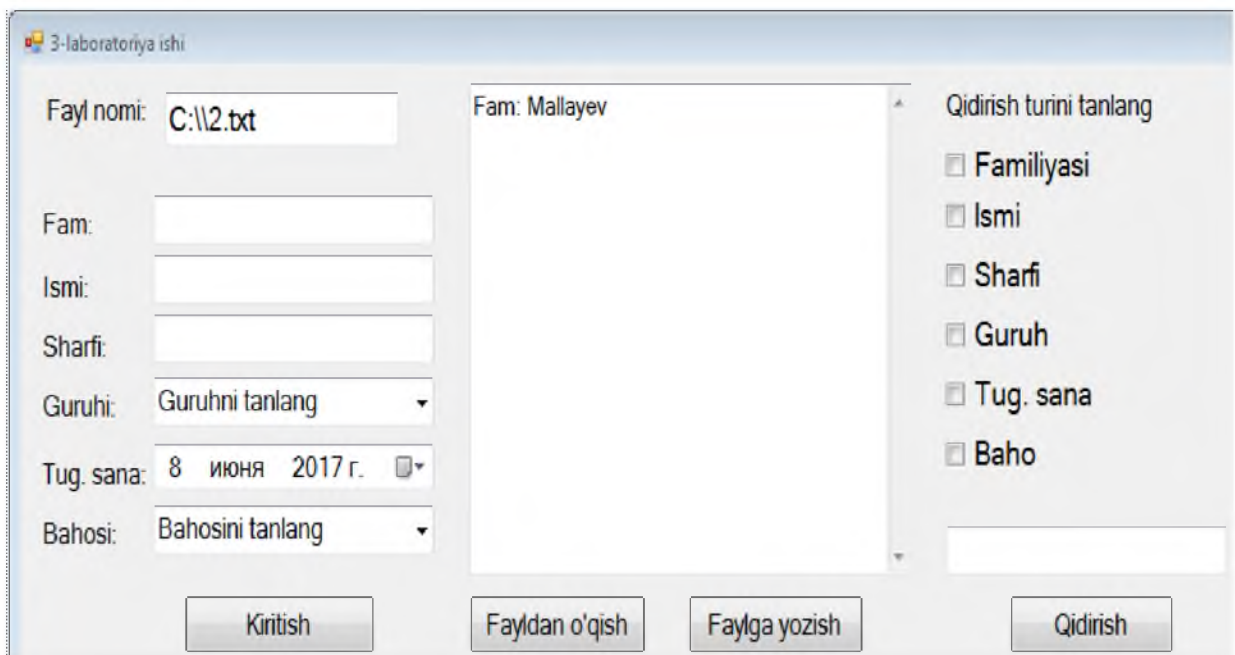
```

```

43. else {qq=0;}}
44. else if(Form1::checkBox6->Checked==true){y=6;
45. try{baho=Convert::ToInt64(textBox6->Text); qq=1;}
46. catch (Exception^ vaziyat){qq=0;
    MessageBox::Show(vaziyat->Message, "Bahoni kiritishda
    xatolik", MessageBoxButtons::OK,
    MessageBoxIcon::Exclamation); } }
47. else {goto bir;}
48. if(qq==1){Tf=gcnew String(fam); int j=Tf->Length-1;
    Tf=Tf->Substring(0,j);
49. Form1::textBox3->AppendText(Tf+System::Environment::
    NewLine); Ti=gcnew String(ism);
50. Form1::textBox3->AppendText(Ti+System::Environment::
    NewLine); Ts=gcnew String(sharfi);
51. Form1::textBox3->AppendText(Ts+System::Environment::
    NewLine); Tg=gcnew String(guruh);
52. Form1::textBox3->AppendText(Tg+System::Environment::
    NewLine); Tu=gcnew String(Tu_sana);
53. Form1::textBox3->AppendText(Tu+System::Environment::
    NewLine); Tb=gcnew
    String(baho.ToString());Form1::textBox3->AppendText(
    Tb+System::Environment::NewLine);
54. e=false;}
55. if(!fgets(temp,2,oqi))break; }
56. if(e){MessageBox::Show("Afsuski siz izlayotgan talaba
    topilmadi","Qidirish natijasi");}}
57. else { bir:MessageBox::Show("Qidiriladigan talaba
    kiriting","Qidirish natijasi");}}
58. private: System::Void button2_Click(System::Object^
    sender, System::EventArgs^ e) {
59. char *satr=and_SysStringToChar(textBox1->Text);
    qidirish(satr);
        1. }

```

4- qadamdan keyin asosiy oyna ko‘rinishi 6.17- rasmdagidek bo‘ladi:



6.17- rasm. Asosiy oynaning 3 ta qismini ko'rishini

5- qadam: Asosiy oynaning fonini o'zgartirish uchun **ListBox** komponentasi joylashtiriladi va sichqoncha o'ng tugmasi ikki marta tez bosilib quyidagi kodlar yoziladi:

```
if (listBox1->Text == "Transparent") return;
this->BackColor = Color::FromName(listBox1->Text);
this->Text = "Rang: " + listBox1->Text;
```

6- qadam: Hisobodni **MS WORD** dasturida ochish uchun **linkLabel** komponentasi joylashtiriladi va sichqoncha o'ng tugmasi ikki marta tez bosilib, quyidagi kodlar yoziladi:

```
String^ myFavoritesPath = Environment::GetFolderPath(
Environment::SpecialFolder::Favorites);
OpenWithStartInfo("winword.exe", textBox1->Text);
```

Dasturning hamma kodlari quyidagicha:

1. **#pragma once**
2. **#include<string.h>**
3. **#include<iostream>**
4. **#include<stdio.h>**
5. **#include<fstream>**
6. **#include"talaba.h"**


```

7. namespace My3lab {
8. using namespace System;
9. using namespace System::ComponentModel;
10. using namespace System::Collections;
11. using namespace System::Windows::Forms;
12. using namespace System::Data;
13. using namespace System::Drawing;
14. using namespace System::Diagnostics;
15. using namespace System::ComponentModel;
16. using namespace System::Runtime::InteropServices;
17. using namespace std;
18. char ss[300]="";
19. public ref class Form1 : public System::Windows::
    Forms::Form{
20. public: Form1(void) { InitializeComponent();      }
21. protected:
22. ~Form1(){
23. if (components){
24. delete components;}  }
25. private: System::ComponentModel::Container
    ^components;
26. #pragma region Windows Form Designer generated code
27. void InitializeComponent(void){      }
28. void OpenWithStartInfo(String^ kamonda, String^
    faylname){ProcessStartInfo^ startInfo = gnew
    ProcessStartInfo(kamonda); startInfo->WindowState =
    ProcessWindowState::Minimized;
29. Process::Start(startInfo); startInfo->Arguments =
    faylname;
30. Process::Start(startInfo);}

31. #pragma endregion
32. private: System::Void
    linkLabel1_LinkClicked(System::Object^ sender,
    System::Windows::Forms::LinkLabelLinkClickedEventArgs^
    e) {String^ myFavoritesPath=
    Environment::GetFolderPath(Environment::SpecialFolder::F
    avorites);
33. OpenWithStartInfo("winword.exe",textBox1->Text);  }
34. char * and_SysStringToChar(System::String^ string){
    return (char*)(void*)Marshal::StringToHGlobalAnsi
    (string); }

```

```

35. // Faylga yozishning antiqa yangi usuli
36. void loadFtomFile(char *FileName, ListBox ^lb){
37. char s[300]; String^ AA;
38. FILE *fp;
39. int i;
40. i=0;
41. lb->Items->Clear();
42. if(!(fp=fopen(FileName,"r"))){ return;}
43. while(!feof(fp)){
44. fgets(s,20,fp); strcat(ss,s,strlen(s));
45. String^ dd=gcnew String(s);
46. int j=dd->Length-1; dd=dd->Substring(0,j); lb->Items-
    >Add(dd); i++;}
47. static int j=1; int y,qq; bool e;
48. void qidirish(char *FileName){ e=true;
49. Form1::textBox3->Clear();
50. if(textBox6->Text!=""){
51. FILE *oqi; oqi=fopen(FileName,"r");
52. if(!oqi){ MessageBox::Show("Xato"); return;}
53. char temp[40]; char ism[40];
54. char fam[40]; char sharfi[40];
55. char guruh[40]; char Tu_sana[40];
56. int baho;
57. String^ Tt,^Ti,^Tf,^Ts,^Tg,^Tu,^Tb;
58. while(!feof(oqi)){
59. int n; fgets(temp,25,oqi); fgets(fam,25,oqi);
60. fgets(ism,25,oqi); fgets(sharfi ,25,oqi);
61. fgets(guruh ,25,oqi); fgets(Tu_sana,25,oqi);
62. fscanf(oqi,"%d",&baho);
63. if(Form1::checkBox1->Checked==true){y=1;
64. char *satr=and_SysStringToChar(textBox6->Text);
65. if(strstr(fam,satr)!=NULL){qq=1;} else {qq=0;} }
66. else if(Form1::checkBox2->Checked==true){y=2; char
    *satr=and_SysStringToChar(textBox6->Text);
67. if(strstr(ism,satr)!=NULL){qq=1;} else {qq=0;} }
68. else if(Form1::checkBox3->Checked==true){y=3; char
    *satr=and_SysStringToChar(textBox6->Text);
69. if(strstr(sharfi,satr)!=NULL){qq=1;} else {qq=0;}}
70. else if(Form1::checkBox4->Checked==true){y=4; char
    *satr=and_SysStringToChar(textBox6->Text);
71. if(strstr(guruh,satr)!=NULL){qq=1;} else {qq=0;}}

```

```

72. else if(Form1::checkBox5->Checked==true){y=5;      char
    *satr=and_SysStringToChar(textBox6->Text);
73. if(strstr(Tu_sana,satr)!=NULL){qq=1;} else {qq=0;}}
74. else if(Form1::checkBox6->Checked==true){y=6;
    try{baho=Convert::ToInt64(textBox6->Text); qq=1;} catch
    (Exception^ vaziyat){qq=0; MessageBox::Show(vaziyat-
    >Message, "Bahoni kiritishda xatolik",
    MessageBoxButtons::OK, MessageBoxIcon::Exclamation);}}
75. else {goto bir;}
76. if(qq==1){
77. Tf=gcnew String(fam); int j=Tf->Length-1;
78. Tf=Tf->Substring(0,j);
79. Form1::textBox3->AppendText(Tf+System::Environment::
    NewLine);
80. Ti=gcnew String(ism);Form1::textBox3->AppendText(Ti+
    System::Environment::NewLine);
81. Ts=gcnew String(sharfi); Form1::textBox3-
    >AppendText(Ts+System::Environment::NewLine);
82. Tg=gcnew String(guruh); Form1::textBox3->AppendText(
    Tg+System::Environment::NewLine
83. Tu=gcnew String(Tu_sana);
84. Form1::textBox3->AppendText(Tu+System::Environment::
    NewLine) Tb=gcnew String(baho.ToString());
    Form1::textBox3->AppendText(Tb+System::Environment::
    NewLine);e=false;}
85. if(!fgets(temp,2,oqi))break; }
86. if(e){MessageBox::Show("Afsuski siz izlayotgan talaba
    topilmadi","Qidirish natijasi");}}
87. else { bir:MessageBox::Show("Qidiriladigan talaba
    kiriting","Qidirish natijasi");}}
88. private: System::Void button1_Click(System::Object^
    sender, System::EventArgs^ e) {
89. textBox3->AppendText(j.ToString()+" -
    talaba"+Environment::NewLine);
90. textBox3->AppendText(textBox2->Text+Environment::
    NewLine);
91. textBox3->AppendText(textBox4->Text+Environment::
    NewLine);
92. textBox3->AppendText(textBox5->Text+Environment::
    NewLine);
93. textBox3->AppendText(comboBox1->Text+Environment::
    NewLine);

```

```

94. textBox3->AppendText(String::Format("{0}",
    dateTimePicker1->Text)+Environment::NewLine);
95. try{
96. textBox3->AppendText(comboBox2->Text+Environment::
    NewLine);}
97. catch(Exception^ vaziyat){ MessageBox::Show("Bahosini
    notogri kiritdingiz"); }j++;}
98. private: System::Void button2_Click(System::Object^
    sender, System::EventArgs^ e) {
99. char *satr=and_SysStringToChar(textBox1->Text);
    qidirish(satr); }
100. String ^ filename;
101. private: System::Void Form1_Load(System::Object^
    sender, System::EventArgs^ e) {
102. textBox3->Clear(); filename = textBox1->Text;
    dateTimePicker1->Format = DateTimePickerFormat::Custom;
    dateTimePicker1->CustomFormat = "dd.dd.yyyy";
    array<String^>^ Rang =
    Enum::GetNames(KnownColor::typeid);
103. listBox1->Items->Clear();
104. listBox1->Items->AddRange(Rang);
105. listBox1->Sorted = true; }
106. private: System::Void button3_Click(System::Object^
    sender, System::EventArgs^ e) {
107. filename = textBox1->Text;
108. try{
109. System::Text::Encoding^ shirift
    =System::Text::Encoding::GetEncoding(1251);
110. IO::StreamReader^ oquvchi = gcnew
    IO::StreamReader(filename, shirift);
111. textBox3->Text = oquvchi->ReadToEnd(); oquvchi-
    >Close(); }
112. catch (IO::FileNotFoundException^ vaziyat){
113. MessageBox::Show(vaziyat->Message + "\nBunday fayl
    yo'q", "Xatolik", MessageBoxButtons:: OK,
    MessageBoxIcon::Exclamation);}
114. catch (Exception^ vaziyat){
115. MessageBox::Show(vaziyat->Message,
    "Xatolik", MessageBoxButtons::OK, MessageBoxIcon::
    Exclamation);} }
116. private: System::Void button4_Click(System::Object^
    sender, System::EventArgs^ e) {

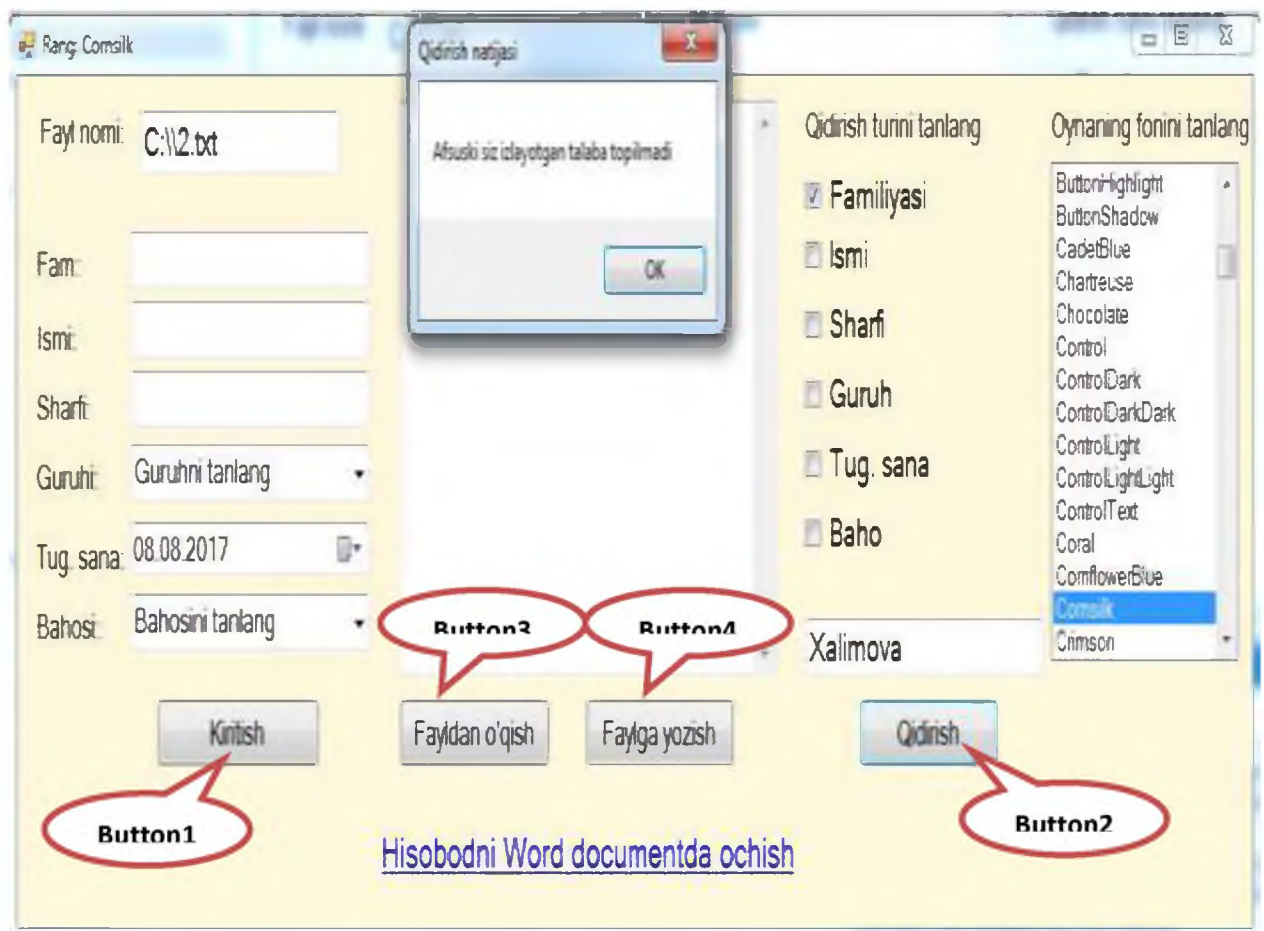
```

```

117. filename = textBox1->Text;
118. try{
119. auto shirift
    =System::Text::Encoding::GetEncoding(1251);
120. auto yozuvch = gcnew IO::StreamWriter(filename,
    false,shirift);
121. yozuvch->Write(textBox3->Text); yozuvch->Close();}
122. catch (Exception^ vaziyat){
123. MessageBox::Show(vaziyat->Message, "Xatolik",
    MessageBoxButtons::OK, MessageBoxIcon:: Exclamation);} }
124. private: System::Void
    listBox1_SelectedIndexChanged(System::Object^ sender,
    System::EventArgs^ e) {
125. if (listBox1->Text == "Transparent") return;
126. this->BackColor = Color::FromName(listBox1->Text);
127. this->Text = "Rang: " + listBox1->Text; }

```

Dasturning umumiy ko‘rinishi quyidagicha:



6.18- rasm. Dastur oynasining umumiy ko‘rinishi

Takrorlash uchun savol va topshiriqlar:

1. Visual Studio qanday dastur?
2. Visual Studio dasturining qanday turlari bor?
3. Visual Studio dasturida qanday dasturlash tillari mavjud?
4. Xossalar va Xodisalar oynasi muharrirning qayerida joylashgan?
5. Visual C++da kiritish va chiqarish operatorlari?
6. Kutubxonalar qanday e'lon qilinadi?
7. O'zgaruvchilarning qanday turlari bor?
8. Console Application muhiti nima?
9. Windows Application muhiti nima?
10. Nomlar fazosi nima uchun ishlatiladi?
11. **Visual C++** da **String** turi qanday tur uning qo'llanilishi qanday bo'ladi?
12. **Visual C++** da boshqa dasturlash tillarining imkoniyatlaridan qaynday foydalanish mumkin?
13. Komponenta nima?
14. Komponentaning qanday xossa va hodisalarini bilasiz?
15. **Visual C++** da turdan turga o'tkazishning nechta turini bilasiz?
16. **Visual Studio** da boshqa dasturlash tillarining kutubxonalari qanday qo'shiladi?
17. **Visual C++** da ma'lumotlarni Form ga chiqarish komponentalarining qanday turlarini bilasiz?
18. **Console Application** muhiti bilan **Windows Application** muhitining farqi nimada?
19. **Visual C++** da global o'zgaruvchilar qanday e'lon qilinadi?
20. Formaning qanday xossalarini bilasiz?

7- LABORATORIYA ISHI

7. KICHIK LOYIHALAR YARATISH

7.1. Matnli maydonga parol kiritish, satrlarga qayta ishlov berish

Ishning maksadi

Button, **TextBox** va **Text** komponentalaridan foydalanib, satrlarga qayta ishlov beruvchi vizual dastur yaratish.

Text komponentasining **PasswordChar** xossasi yordamida parollarni kiritishni boshqa shirifflarga o'zgartirish mumkin. Ya'niy **text** xossasidan kiritilayotgan ma'lumotlarning ko'rinishlarini o'zgartirib, foydalanuvchi tanimaydigan shirifflarda kiritilishini ta'minlaydi. Bu xususiyat dastur xavfsizligini ta'minlashda katta ahamiyat kasb etadi. Ushbu laboratoriya ishida dasturga parol yordamida kirish jarayonlarini boshqaruvchi va virtual tarzda komponentalar yaratuvchi amaliy vizual dastur yaratiladi. Ushbu jarayonlarni boshqarishda global e'lon qilinmagan tugmalarni bosgandagi hodisalariga ham to'xtalib o'tiladi. Asosiy maqsad **char** turidagi massivlar bilan ishlashda vujudga keladigan muammoli holatlarni bartaraf qilish usullarini o'rganishdir. Masalan **Text** komponentasidan kiritilayotgan ma'lumotlar **String** tipida. Bunday tipdagi ma'lumotlarni **char** tipiga o'tkazish. Yoki bo'lmasam **char** tipidagi massivga o'tkazishni ko'rib chiqamiz. 17- jadvaldagi masalalarni yechishda quyidagi keltirilgan namunaga ahamiyat berish zarur.

Topshiriq

Ma'ruza va amaliyot darslarida olingan ma'lumotlarga asoslanib, o'z variantingiz uchun quyidagilarni bajarinig:

- a) 13- jadvaldan o'z variantingizni ko'chirib oling;
- b) quyidagi namuna asosida o'z variantingizdagi masalalarni dasturini tuzing.

13- jadvaldagi har bir variantda quyidagilar bajarilishi lozim:

- 1- dasturga kirishda parol so'ralsin;

- 2- yangi parol kiritish va saqlash imkoniyatlari bo‘lsin;
- 3- dasturni yangidan ishga tushirilganda yangi kiritilgan parol saqlanishi lozim.

13- jadval

1. Matnli faylda nechta so‘z va gap qatnashayotganligini aniqlovchi dastur tuzing.		
	Kiruvchi	Chiquvchi
	Bugun burunchi para C++ darsi. 2- para Matematika.	7 2
	Toshkent O‘zbekiston poytaxti.	3 1
	Mening Malibu rusumli mashinam bor edi. Uni sotdim.	8 2
2. Matnli faylda nechta “a” va “.” qatnashayotganligini aniqlovchi dastur tuzing.		
	Kiruvchi	Chiquvchi
	Mana senga olam olam gul.	5 1
	.Toshkent O‘zbekiston poytaxti.	1 2
	Mening Malibu rusumli mashinam bor.	3 1
3. Berilgan gapdagi simvollar soni n tadan kam bo‘lgan so‘zlarni o‘chiring.		
	Kiruvchi	Chiquvchi
	5 Mana senga olam olam gul.	senga
	8 Toshkent O‘zbekiston poytaxti.	O‘zbekiston
	7 Mening Malibu rusumli mashinam bor.	rusumli mashinam
4. Matnli satrda katta harflar bilan berilgan satr berilgan. Undagi qism satr necha marta qatnashayotganligini aniqlovchi dastur tuzing.		
	Kiruvchi	Chiquvchi
	DEF DEFDEFABCABCZABCDE FDEF	4
5. Matnli fayldagi simvollar ketma-ketligi bir nechta probellar bilan ajratilgan holda berilgan. Undagi eng uzun so‘zning “a” harflarini “b” bilan almashtiring.		
	Kiruvchi	Chiquvchi

	Mana senga olam olam gul!	Mana sengb olam olam gul!	
6. Matnli fayldagi simvollar ketma-ketligi bir nechta probellar bilan ajratilgan holda berilgan. Undagi eng uzun va eng kalta soʻzlarni sonini probel bilan chiqaring. Masalan, “hi! say biggest” uchun natija “1 2”			
	Kiruvchi	Chiquvchi	
	Mana senga olam olam gul!	1 1	
7. Matnli fayldan ortiqcha probellarni olib tashlovchi dastur tuzing. Har bir soʻz orasida bitta probel tursin.			
	Kiruvchi	Chiquvchi	
	Mana senga olam olam gul!	Mana senga olam olam gul!	
8. Matnli fayldagi gaplarni birinchini ikkinchiga, uchinchini toʻrtinchiga va hokazo soʻzlarini oʻzgartiruvchi dastur tuzing.			
	Kiruvchi	Chiquvchi	
	Mana senga olam olam gul! Havo yaxshi.	senga Mana olam olam gul! yaxshi Havo.	
9. Matnli fayldagi birinchini ikkinchiga, uchinchini toʻrtinchiga va hokazo gaplarni oʻrnini oʻzgartiruvchi dastur tuzing.			
	Kiruvchi	Chiquvchi	
	Mana senga olam olam gul! Havo yaxshi.	Havo yaxshi. Mana senga olam olam gul!	
10. Matndagi eng koʻp qatnashgan harfni topuvchi dastur tuzing. Sonini va harflarni matnda berilgan tartib boʻyicha chiqaring.			
	Kiruvchi	Chiquvchi	
	Mana senga olam olam gul! Havo yaxshi.	7 a	
	Bugun imtixon. Demak, bayram ekan.	4 a	
11. Matndagi eng kam qatnashgan harflarni topuvchi dastur tuzing. Sonini va harflarni matnda berilgan tartib boʻyicha chiqaring.			
	Kiruvchi	Chiquvchi	
	Mana senga olam olam gul! Havo yaxshi.	1 n i x y	
	Bugun imtixon. Demak, bayram ekan.	1 g t x o D y r	
12. Soʻz berilgan. Shu soʻzni palindromlikka tekshiring. Boshidan va teskarisiga oʻqilganda bir xil boʻlsa bu palindrom boʻladi. Masalan, aziza. Dasturingiz kiritilgan soʻz palindrom boʻlsa “Yes”, aks holda “No” chiqarsin.			

	Kiruvchi	Chiquvchi
	kiyik	Yes
	aziza	Yes
	aziz	No
13. Berilgan soʻzga nechta harf qoʻshsa palindrom boʻladi. Shu sonni chiqaring. Masalan,		
	Kiruvchi	Chiquvchi
	aziz	1
	olim	3
	aslo	3
14. Berilgan soʻz palindrom boʻlishi uchun qanday boʻlishi kerak. Masalaga eʼtibor bering (1354).		
	Kiruvchi	Chiquvchi
	Aziz	AzizA
	Olim	OlimilO
	OnLine	OnLineniLnO
	KiyiK	KiyiK
15. Matnli fayldagi soʻzlarni ulardagi simvollar soni boʻyicha kamayish va oʻsish tartibida chiqaring. Bunda satr uzunligi 200 va har bir soʻzning uzunligi 10 boʻlishi mumkin. Masalan,		
	Kiruvchi	Chiquvchi
	2014 yil Sogʻlom bola yili	Sogʻlom 2014 bola yili yil
16. Matnli fayldagi soʻzlar berilgan. Shu soʻzlardagi simvollar soni beshdan oshganlarini oxirgi uchta simvolini “\$” bilan almashtiring. Masalan,		
	Kiruvchi	Chiquvchi
	big	big
	mouses	mou\$
	industry	indust\$
	dastur	das\$
	klaviatura	klaviat\$
17. Matnli fayldagi soʻzlardan faqatgina boshi va oxiridagi simvollarini bir xil boʻlmagan sozlarni qoldiring (Katta va kichik harflar farq qilmaydi). Agarda bunday soʻz boʻlmasa “NO” chiqaring.		
	Kiruvchi	Chiquvchi
	Mana senga olam olam gul!	NO

	Toshkent O‘zbekiston poytaxti.	O‘zbekiston poytaxti.	
	Mening Malibu rusumli mashinam bor.	Mening Malibu rusumli bor.	
18. Berilgan so‘zdagi katta va kichik harflar nisbatini toping. Avval katta harflarni keyin kichik harflarni foizini nuqtadan keyin ikkita aniqlikda chiqaring.			
	Kiruvchi	Chiquvchi	
	Ola!	25.00 50.00	
	Salom	20.00 80.00	
	Toshkent	25.00 75.00	
	Toshkent	12.50 87.50	
19. Berilgan matndagi so‘zlarning o‘rtacha uzunligini toping.			
	Kiruvchi	Chiquvchi	
	Mana senga olam olam gul	4.00	
	Toshkent O‘zbekiston poytaxti	9.00	
	Tashkent capital of Uzbekistan	6.75	
20. “One” va “puton” bir-biri bilan gaplashyapdi. “One” "out" va "output" so‘zlarini va “puton” so‘zini gapira oladi. “Puton” esa "in", "input" va "one" so‘zlarini gapira oladi. Va ular bir birini probelsiz yozilgan ushbu so‘zlar bilan bemalol tushunishadi. Sizga N ta satrda ularning dialogi berilgan. Shu dialoglardan qaysi biri dialog bo‘lishini aniqlang. Agar berilgan dialog faqat “one” va “puton” ning so‘zlaridan iborat bo‘lsa “YES” aks holda “NO” chiqaring (1102). Masalan,			
	Kiruvchi	Chiquvchi	
	6	YES	
	puton	NO	
	inonputin	YES	
	oneputonininputoutoutput	NO	
	oneininputwooutput	NO	
	outpu	NO	
	utput	NO	
21. Siz maqolalarni qabul qilyapsiz. Maqolalardagi senzura oid so‘zlarni tekshirishingiz kerak. Berilgan senzura oid so‘zlarni maqoladan qidiring. Birinchi satrda n ($1 \leq n \leq 10000$) soni berilgan. Keyingi n ta satrda esa senzura oid so‘zlar berilgan. Undan keyin esa maqoladagi satrlar soni m berilgan. Birinchi marta topilgan senzura oid so‘zni satrini va o‘rnini chiqaring. Agar bunday so‘zlar topilmasa, “NO” chiqaring (1269). Masalan,			

	Kiruvchi	Chiquvchi
5 dear sweetie angel dream baby 8	Had I the heavens' embroidered cloths, Enwrought with golden and silver light, The blue and the dim and the dark cloths Of night and light and the half-light, I would spread the cloths under your feet: But I, being poor, have only my dreams; I have spread my dreams under your feet; Tread softly because you tread on my dreams.	6 33
22. Berilgan matndagi takrorlanuvchi soʻzlarni olib tashlang.		
	Kiruvchi	Chiquvchi
	Mana senga olam olam gul.	Mana senga olam gul.
	I am going at home. They are going at home too.	I am going at home. They are home too.
23. Berilgan matndagi juft simvolli soʻzlarning birinchi yarmini chiqaring.		
	Kiruvchi	Chiquvchi
	Mana senga olam olam gul.	Ma senga ol ol gul.
	I am going at home. They are going at home too.	I a going a ho. Th are going t ho too.
24. Berilgan matndagi juft simvolli soʻzlarning ikkinchi yarmini chiqaring.		
	Kiruvchi	Chiquvchi
	Mana bugun bayram.	na bugun ram.
	I am going at home. They are going at home too.	I m going t me. ey are t me too.
25. Berilgan matndagi juft simvolli soʻzlarning birinchi va ikkinchi yarimlarini oʻrnini almashtiring.		
	Kiruvchi	Chiquvchi
	Mana bugun bayram.	naMa bugun rambay.
	I am going at home. They are going at home too.	I ma going ta meho. eyTh are ta meho too.

26. Berilgan matndagi juft simvolli soʻzlarning birinchi va ikkinchi yarimlarini oʻrnini almashtiring.		
	Kiruvchi	Chiquvchi
	Mana bugun bayram.	naMa bugun rambay.
	I am going at home. They are going at home too.	I ma going ta meho. eyTh are ta meho too.
27. Berilgan matndagi toq simvolli soʻzlarni teskari tartibda chiqaring.		
	Kiruvchi	Chiquvchi
	Mana bugun bayram.	Mana nugub bayram
	I am going at home. They are going at home too.	I am gniog at home. They era gniog at home oot.
28. Berilgan matnda necha xil unli tovush qatnashayotganligini aniqlovchi dastur tuzing.		
	Kiruvchi	Chiquvchi
	Mana bugun bayram.	2
	Bugun rosa qiyin kun boʻldi.	4
29. Berilgan matnda necha xil undosh tovush qatnashayotganligini aniqlovchi dastur tuzing. Katta va kichik harflar farq qilmaydi.		
	Kiruvchi	Chiquvchi
	Mana bugun bayram.	6
	Bugun rosa qiyin kun boʻldi.	10
30. Berilgan matnda barcha probellarni “ ” bilan almashtiring.		
	Kiruvchi	Chiquvchi
	Mana bugun bayram.	Mana_bugun_bayram.
	Bugun rosa qiyin kun boʻldi.	Bugun_rosa_qiyin_kun__boʻldi.

13- jadvaldagi topshiriqlarni bajarish uchun uslubiy qoʻllanmalar.
Visual C++ dasturida 30 – variantdagi masalani Windows Application muhitida ishlanishi:

1- qadam. Formaga 2 ta **TextBox**, 2 ta **label**, 1 ta **Button** komponentalari joylashtiriladi. **Button** komponentasini **onclick** hodisasiga quyidagi kodlar yoziladi:

1. `char satr[200]="";`
2. `for(int i=0; i<textBox1->Text->Length; i++){`
3. `satr[i]=textBox1->Text[i];}`
4. `for(int i=0; i<strlen(satr); i++)`

```

5. { if(satr[i]==‘ ‘){satr[i]=‘_‘;}}
6. String^ s=gcnew String(satr);
7. textBox2->Text=s;

```

2- qadam. Form1 ning load hodisasiga quyidagi kodlar yoziladi:

```

1. label2->Visible=false;
2. label1->Text="Parolni kiriting";
3. textBox2->Visible=false;
4. button1->Visible=false;
5. Button^ button4 = gcnew Button();//button yaratish
6. button4->Font = gcnew System::Drawing::Font("Courier
New", 16.0F, FontStyle::Italic);
7. textBox1->PasswordChar = ‘*‘;//kodlar terilganda *
//ko‘rinishida bo‘lishini ta‘minlaydi
8. textBox1->Font = gcnew System::Drawing::Font("Courier
New",16.0F);
9. textBox1->Size = Drawing::Size(208, 35);
10. Form1::Height=200;
11. Form1::Width=230;
12. button4->Visible = true;
13. button4->Size = Drawing::Size(100, 30);
14. button4->Location = Drawing::Point(50, 90);
15. button4->Text = "Ok";
16. this->Controls->Add(button4);
17. parol="Salom";
18. button4->Click += gcnew EventHandler(this,
&Form1::UelchokNaKnopke );

```

3- qadam. Satrlarga ishlov beruvchi funksiyalar kutubxonasi e‘lon qilinadi va UelchokNaKnopke funksiyasi tuziladi:

```

1. #pragma once
2. #include<string.h>
3. private: System::Void UelchokNaKnopke(System::Object^
sender, System::EventArgs^ e){ newparol=textBox1-
>Text;
4. if(parol==newparol){
5. textBox1->Size = Drawing::Size(208, 242);
6. Form1::Size=Drawing::Size(503,369);
7. textBox1->Location = Drawing::Point(20, 35);
8. textBox1->Text = ""; textBox1->PasswordChar = NULL;
9. label1->Text="Kiruvchi ma‘lumotlar";

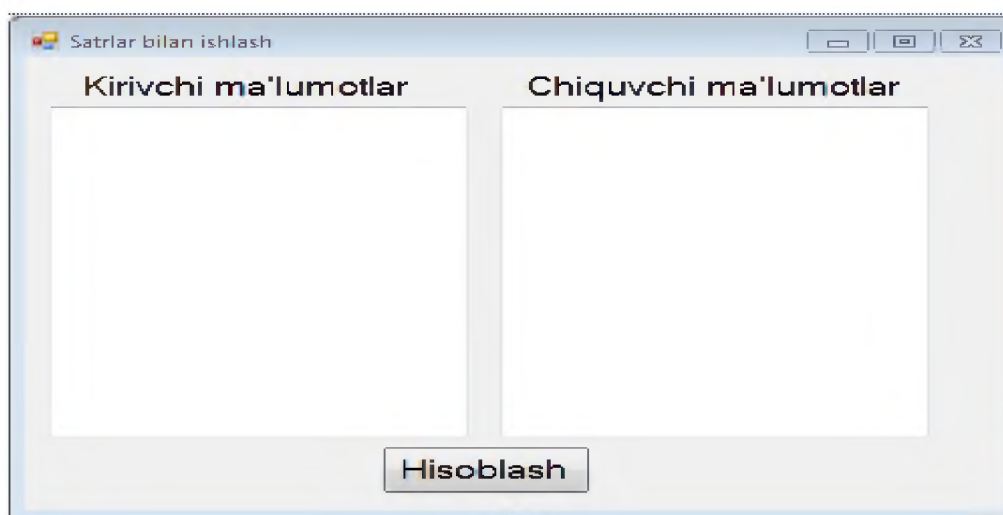
```

```

10. label1->Visible=true; label2->Visible=true;
11. textBox1->Visible=true; textBox2->Visible=true;
12. button1->Visible=true;}
13. else {
14. System::Windows::Forms::FormClosingEventArgs^ r;
15. auto MBox = MessageBox::Show("Yana urunib ko'rasmi?",
    "Xato", MessageBoxButtons::YesNoCancel,
    MessageBoxIcon::Exclamation);
16. // YES – dialog; NO – выход; CANCEL - redaktirovat
17. if (MBox == Windows::Forms::DialogResult::No) return;
18. if (MBox == Windows::Forms::DialogResult::Cancel){
    this->Close();}
19. if (MBox == Windows::Forms::DialogResult::Yes)
20. {   textBox1->Text="";
21. }   }   }

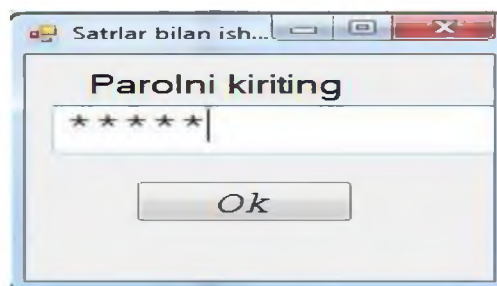
```

Dasturning umumiu ko'rinishi 5.1-rasmda keltirilgan.



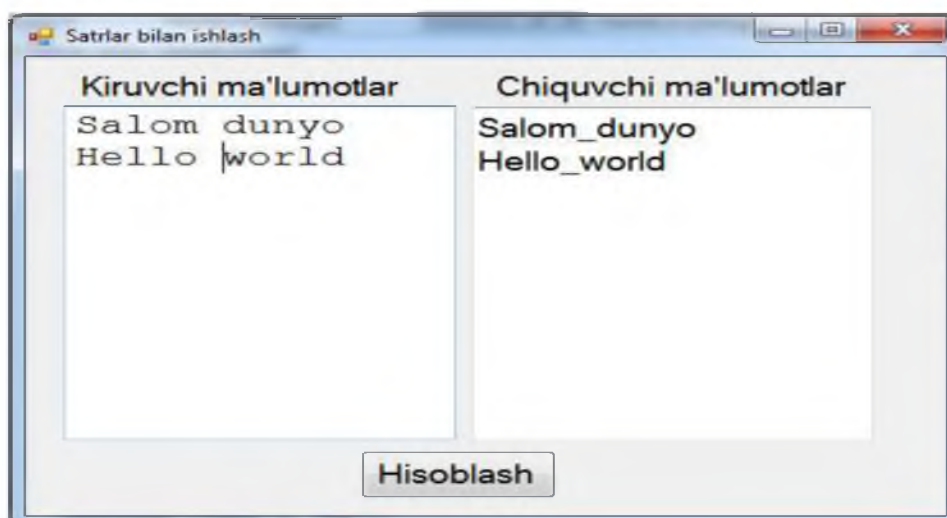
5.1-rasm. Dasturning umumiy ko'rinishi

Lekin dastur ishga tushganda **Form1** ning o'lchami o'zgaradi. Ya'niy foydalanuvchidan parol kiritishni talab qiluvchi oynaga o'zgaradi. Uning ko'rinishi 5.2- rasmda keltirilgan.



7.2-rasm. Dasturning dastlabki ko‘rinishi

Bu oynada **OK** tugmachasi virtual yaratilgan va parol kiritish uchun **TextBox** komponentasining **PasswordChar** xossasi = ‘*‘ ga tenglashtirilgan. Parol to‘g‘ri terilgandan keyin dasturning ishlashi quyidagicha:



7.3-rasm. Dasturning parol to‘g‘ri kiritilgandan keyingi ko‘rinishi

Dasturning umumiy kodlari quyidagicha:

```

5. #pragma once
6. #include<string.h>
7. namespace Satrlar_bilan_ishlash
8. {
9. using namespace System;
10. using namespace System::ComponentModel;
11. using namespace System::Collections;
12. using namespace System::Windows::Forms;
13. using namespace System::Data;
14. using namespace System::Drawing;

```



```

15. public ref class Form1:public System::Windows::
    Forms::Form
16. {
17. public:
18. Form1(void){
19. InitializeComponent(); }
20. protected:
21. ~Form1(){
22. if (components){
23. delete components;}          }
24. private: System::Windows::Forms::Button^  button1;
25. protected:
26. private: System::Windows::Forms::TextBox^  textBox1;
27. private: System::Windows::Forms::Label^  label1;
28. private: System::Windows::Forms::TextBox^  textBox2;
29. private: System::Windows::Forms::Label^  label2;
30. private:
31. System::ComponentModel::Container ^components;

32. #pragma region Windows Form Designer generated code
33. void InitializeComponent(void){
34. // komponentalar e'lon qilinadigan bo'limi}
35. #pragma endregion
36. private: System::Void button1_Click(System::Object^
    sender, System::EventArgs^  e) {
37. char satr[200]="";
38. for(int i=0; i<textBox1->Text->Length; i++)
39. { satr[i]=textBox1->Text[i]; }
40. for(int i=0; i<strlen(satr); i++) { if(satr[i]== ' ')
    {satr[i]='_';}}
41. String^ s=gcnew String(satr); textBox2->Text=s;}
42. String^ parol, ^newparol;
43. private: System::Void Form1_Load(System::Object^
    sender, System::EventArgs^  e) {
44. label2->Visible=false; label1->Text="Parolni kiriting";
45. textBox2->Visible=false; button1->Visible=false;
46. Button^ button4 = gcnew Button();
47. button4->Font = gcnew System::Drawing::Font("Courier
    New", 16.0F, FontStyle::Italic);
48. textBox1->PasswordChar = '*';
49. textBox1->Font = gcnew System::Drawing::Font("Courier
    New",16.0F);

```

```

50. textBox1->Size = Drawing::Size(208, 35);
51. Form1::Height=200;
52. Form1::Width=230;
53. button4->Visible = true; button4->Size =
    Drawing::Size(100, 30);
54. button4->Location = Drawing::Point(50, 90); button4-
    >Text = "Ok";
55. this->Controls->Add(button4); parol="Salom";
56. button4->Click += gcnew EventHandler(this,
    &Form1::UelchokNaKnopke );
57. }
58. private: System::Void UelchokNaKnopke(System::Object^
    sender, System::EventArgs^ e)
59. {
60.     newparol=textBox1->Text;
61.     if(parol==newparol)
62.     {
63.         textBox1->Size = Drawing::Size(208, 242);
64.         Form1::Size=Drawing::Size(503,369);
65.         textBox1->Location = Drawing::Point(20, 35);
66.         textBox1->Text = "";
67.         textBox1->PasswordChar = NULL;
68.         label1->Text="Kiruvchi ma'lumotlar";
69.         label1->Visible=true;
70.         label2->Visible=true;
71.         textBox1->Visible=true;
72.         textBox2->Visible=true;
73.         button1->Visible=true;
74.     }
75.     else
76.     {
77.         System::Windows::Forms::FormClosingEventArgs^ r; auto
            MBox = MessageBox::Show("Yana urunib ko'rasmi?", "Xato",
            MessageBoxButtons::YesNoCancel,
            MessageBoxIcon::Exclamation);
78.         // YES – dialog; NO – выход; CANCEL - redaktirovat
79.         if (MBox == Windows::Forms::DialogResult::No) return;
80.         if (MBox == Windows::Forms::DialogResult::Cancel){ this-
            >Close();}
81.         if (MBox == Windows::Forms::DialogResult::Yes)
82.         {
83.             textBox1->Text="";
84.         }
85.     }
86. }; }

```

7.2. Ko'p oynali muhitlar bilan ishlash. Fayl ochish va saqlash. Formaning Closing hodisasi

Ishning maksadi

TextBox, MenuStrip, OpenFileDialog va **SaveFileDialog** komponentalaridan foydalanib, matnli va **rtf** kentgaytmali fayllarga qayta ishlov beruvchi vizual dastur yaratish.

Bizga ma'lumki matnli fayllarga ma'lumotlarni yozuvchi va o'quvchi maxsus funksiyalar mavjud. Shunday bo'lsa ham **Visual C++** muharririning ham maxsus shunday funksiyalari mavjud. Ushbu funksiyalarni vizual komponentalar bilan ishlatib, kichik matn muharririni yaratish mumkin. Ushbu laboratoriya ishida fallarga ishlov berishning bir necha usullari bilan tanishib chiqiladi.

Topshiriq

Ma'ruza va amaliyot darslarida olingan ma'lumotlarga asoslanib, o'z variantingiz uchun quyidagilarni bajarinig:

- a) 14- jadvaldan o'z variantingizni ko'chirib oling;
- b) quyidagi namuna asosida o'z variantingizdagi masalalarni dasturini tuzing.

Har-bir variantning 1- formasida 2 ta masalani yechish dasturini tuzing.

14- jadval

№	Masala sharti
1.	1. N va K butun musbat sonlar va fayl nomi berilgan. Yangi matnli fayl hosil qilinsin va unga N ta satr va har bir satr K ta "*" (yulduzcha) belgisidan iborat bo'lsin. 2. 10 ta raqamdan iborat S satr va shifrlangan matnli fayl berilgan. Sezar usulidagi algoritim bo'yicha shifrlangan matni deshifrovchi programma tuzilsin. 2-formada matematika fanidan 2 ta variantli test dasturini tuzing.
2.	1. N ($0 < N < 27$) butun son va fayl nomi berilgan. Berilgan nomdagi matnli fayl hosil qilinsin va unga: birinchi satri "a" kichik lotin harfi, ikkinchisiga "ab", uchinchisiga "abc" va h.k satrlarni saqlovchi N ta satr yozilsin. 2. 10 ta raqamdan iborat S satr va lotin harflaridan iborat matnli fayl berilgan. Matnli fayli quyidagicha shifrlang: Matnli fayl satrining K - belgisini, shu

	<p>belgining kodiga S satridagi K -raqamini qo‘shishdan hosil bo‘lgan kod belgisiga almashtiring. Agar $K=11$ bo‘lsa, ya‘na satrning birinchi raqamidan boshlang.</p> <p>2-formada fizika fanidan 3 ta variantli test dasturini tuzing.</p>
3.	<p>1. N ($0 < N < 27$) butun son va fayl nomi berilgan. Berilgan nomdagi fayl hosil qilinsin va unga uzunligi N ga teng bo‘lgan N ta satr quyidagicha yozilsin; K-nomerdagi satr ($K=1N$) katta lotin harflarining boshlang‘ich K ta harfini va undan o‘ngda "*" belgisidan iborat bo‘lsin. Masalan $N=4$ uchun fayl quyidagi satrlardan iborat bo‘lishi kerak. "A***", "AB**", "ABC*", "ABCD".</p> <p>2. Matnli fayl berilgan. Undagi har bir uchragan kichik lotin harflarining uchrashlar miqdori sanalsin va tarkibi quyidagi ko‘rinishda bo‘lgan matnli fayl hosil qilinsin "<harf>-<uchrashlar soni>"(masalan, "a-25"). Matnda uchramagan harflar hisobga olinmasin. Satrlarni harflarning uchrash sonini kamayish bo‘yicha, teng sondagi uchrashlarni esa ularning kodlari bo‘yicha o‘shish tartibida joylashtirilsin.</p> <p>2-formada Kimyo fanidan 4 ta variantli test dasturini tuzing.</p>
4.	<p>1. Matnli fayl berilgan. Uning tarkibiga kiruvchi satrlar va belgilar soni (miqdori) chop qilinsin. (satrning oxiri EOLN va EOF fayl oxirlari markerlari belgilarni sanayotganda hisobga olinmasin).</p> <p>2. Matnli fayl berilgan. Undagi har bir uchragan kichik lotin harflarining uchrashlar miqdori sanalsin va tarkibi quyidagi ko‘rinishda bo‘lgan matnli fayl hosil qilinsin "<harf>-<uchrashlar soni>"(masalan, "a-25"). Matnda uchramagan harflar hisobga olinmasin. Satrlar kodi bo‘yicha o‘shish tartibida joylashsin.</p> <p>2-formada Ona tili fanidan 2 ta variantli test dasturini tuzing.</p>
5.	<p>1. Satr va matnli fayl berilgan. S satr fayl oxiriga qo‘shilsin.</p> <p>2. Matnli fayl berilgan. Matnli fayldagi barcha uchragan belgilarni, probel va tinish belgilariga ega bo‘lgan (takrorlanishsiz) belgili fayl hosil qilinsin. Belgilar kodi bo‘yicha kamayish tartibida joylashsin.</p> <p>2-formada Ingliz fanidan 3 ta variantli test dasturini tuzing.</p>
6.	<p>1. Ikkita matnli fayl berilgan. Birinchi fayl oxiriga ikkinchi fayl qo‘shilsin.</p> <p>2. Matnli fayl berilgan. Matnli fayldagi barcha uchragan belgilarni, probel va tinish belgilariga ega bo‘lgan (takrorlanishsiz) belgili fayl hosil qilinsin. Belgilar kodi bo‘yicha o‘shish tartibida joylashsin.</p> <p>2-formada Nemis tili fanidan 4 ta variantli test dasturini tuzing.</p>
7.	<p>1. S satr va matnli fayl berilgan. S satr fayl boshiga qo‘shilsin.</p> <p>2. Matnli fayl berilgan. Matnli fayldagi barcha uchragan belgilarni, probel va tinish belgilariga ega bo‘lgan (takrorlanishsiz) belgili fayl hosil qilinsin. Belgilar matnda birinchi joylashgan tartibida joylashtirilsin.</p> <p>2-formada rus tili fanidan 2 ta variantli test dasturini tuzing.</p>
8.	<p>1. Ikkita matnli fayl berilgan. Birinchi fayl boshiga ikkinchi fayl qo‘shilsin.</p> <p>2. Matnli fayl berilgan. Matnli fayldagi barcha uchragan tinish belgilariga ega bo‘lgan belgili fayl hosil qilinsin.</p>

	2-formada Jismoniy tarbiya fanidan 3 ta variantli test dasturini tuzing.
9.	<p>1. K butun soni va matnli fayl berilgan. K- nomerdagi satrdan oldin bo‘sh satr qo‘yilsin. Agar bunday nomerli satr mavjud bo‘lmasa, u holda fayl o‘zgartirishsiz qoldirilsin.</p> <p>2. Butun sonlardan iborat uchta ustunli jadvalga ega bo‘lgan matnli fayl berilgan. Jadvaldagi har bir ustunining boshiga va oxitiga hamda ular orasiga ajratuvchi belgi joylashtirilgan. Jadvaldagi ustunlar kengligi va ularning tekislanishi hamda ajratuvchi belgilar ko‘rinishi ixtiyoriy. Boshlang‘ich jadvalning har bir satridagi sonlar yig‘indisiga ega bo‘lgan yangi butun sonlar fayli hosil qilinsin.</p> <p>2-formada Axborot xavfsizligi fanidan 4 ta variantli test dasturini tuzing.</p>
10.	<p>1. K butun soni va matnli fayl berilgan. K- nomerdagi satrdan keyin bo‘sh satr qo‘yilsin. Agar bunday nomerli satr mavjud bo‘lmasa, u holda fayl o‘zgartirishsiz qoldirilsin.</p> <p>2. Haqiqiy sonlardan iborat uchta ustunli jadvalga ega bo‘lgan matnli fayl berilgan. Ustun kengligi, tekislash usullari ixtiyoriy ravishda berilgan, maxsus ajratuvchi belgilarga ega emas. Uchta haqiqiy sonlar fayli hosil qilinsin va har bir fayl jadvalning mos ustunidagi sonlarni o‘zida saqlasin.</p> <p>2-formada S++da dasturlash fanidan 2 ta variantli test dasturini tuzing.</p>
11.	<p>1. Matnli fayl berilgan. Undagi barcha bo‘sh satrlar ikkilantirilsin.</p> <p>2. Matnli fayl berilgan. Har bir satrning birinchi 30 ta belgisi matndan, qolgani esa haqiqiy sonlardan iborat. Boshlang‘ich faylning barcha matn qismiga ega bo‘lgan matnli fayl va boshlang‘ich faylning barcha haqiqiy sonlar qismiga ega bo‘lgan haqiqiy sonlar fayli hosil qilinsin.</p> <p>2-formada Mexnat ta‘lim fanidan 3 ta variantli test dasturini tuzing.</p>
12.	<p>1. S satr va matnli fayl berilgan. Fayldagi barcha bo‘sh satrlar S satrga o‘zgartirilsin.</p> <p>2. Butun sonlar fayli va matnli fayl berilgan. Matn faylining har bir satrining oxiriga butun sonlar faylidagi mos sonlari joylashtirilsin. Agar butun sonlar fayli matn faylidan qisqa bo‘lsa, u holda matn faylidagi qolgan satrlar o‘zgartirmasdan qoldirilsin.</p> <p>2-formada Java dasturlash fanidan 4 ta variantli test dasturini tuzing.</p>
13.	<p>1. Bo‘sh bo‘lmagan matnli fayl berilgan. Undagi birinchi satr o‘chirilsin.</p> <p>2. Har bir satrida probellar bilan ajratilgan bir nechta sonlarni tasvirlovchi matnli fayl berilgan (haqiqiy sonlar nol bo‘lmagan kasr qismiga ega). Tarkibida boshlang‘ich faylning barcha butun sonlariga ega bo‘lgan butun sonlar fayli hosil qilinsin.</p> <p>2-formada Falsafa fanidan 2 ta variantli test dasturini tuzing.</p>
14.	<p>1. Bo‘sh bo‘lmagan matnli fayl berilgan. Undagi oxirgi satr o‘chirilsin.</p> <p>2. Har bir satri o‘ng va chap tomonidan bir nechta probellar bilan to‘ldirilgan butun yoki haqiqiy sonlarni tasvirlovchi matnli fayl berilgan. (haqiqiy sonlar nol bo‘lmagan kasr qismiga ega). Butun sonlar miqdori va ularning yig‘indisi chop qilinsin.</p>

	2-formada UZDJQ fanidan 3 ta variantli test dasturini tuzing.
15.	<p>1. K butun soni va matnli fayl berilgan. Undaki K-nomerdagi satr o‘chirilsin. Agar faylda bunday nomerdagi satr mavjud bo‘lmasa, u holda fayl o‘zgartirishsiz qoldirilsin.</p> <p>2. Har bir satrida probellar bilan ajratilgan bir nechta sonlarni tasvirlovchi matnli fayl berilgan (haqiqiy sonlarning kasr qismi noldan farqli). Boshlang‘ich faylning barcha noldan farqli kasr qismiga ega bo‘lgan sonlaridan iborat (taribini o‘zgartirmagan holda) haqiqiy sonlar fayli hosil qilinsin.</p> <p>2-formada Algoritm lash fanidan 4 ta variantli test dasturini tuzing.</p>
16.	<p>1. Matnli fayl berilgan. Undagi barcha bo‘sh satrlar o‘chirilsin.</p> <p>2. Har bir satrida bittadan butun yoki haqiqiy son bo‘lgan matnli fayl berilgan. Sonlar o‘ng va chap tomonidan bir nechta probellar bilan to‘ldirilgan. (haqiqiy sonlarning kasr qismi noldan farqli). Kasr qismi nol bo‘lmagan sonlar miqdori va ularning yig‘indisi chop qilinsin.</p> <p>2-formada Kompyuter grafikasi fanidan 2 ta variantli test dasturini tuzing.</p>
17.	<p>1. Ikkita matnli fayl berilgan. Birinchi faylning har bir satridan so‘ng ikkinchi fayldagi mos satrlar qo‘shilsin. Agar ikkinchi fayl birinchi fayldan kalta bo‘lsa, u holda qolgan satrlar o‘zgartirishsiz qoldirilsin.</p> <p>2. Har bir satrida bittadan butun son bo‘lgan matnli fayl berilgan. Butun sonlar o‘ng va chap tomonidan bir nechta probellar bilan to‘ldirilgan. Shu sonlar miqdori va ularning yig‘indisi chop qilinsin.</p> <p>2-formada Axborot texnologiyalari fanidan 3 ta variantli test dasturini tuzing.</p>
18.	<p>1. K butun soni va matnli fayl berilgan. Faylning har bir satridan birinchi k ta belgi o‘chirilsin. (agar satr uzunligi K dan kichik bo‘lsa, u holda satrning hamma belgilari o‘chirilsin).</p> <p>2. N butun soni va A, V haqiqiy sonlari berilgan. $\sin(x)$ va $\cos(x)$ funksiyasining $[A, B]$ oraliqdagi $(B-A)/N$ qadam bilan hosil bo‘luvchi qiymatlari jadvalini saqlovchi matnli fayl hosil qilinsin. Jadval uchta ustundan iborat: x argumentli (8 ta pozitsiya va uning 4 tasi kasr qismi) va $\sin(x)$ hamda $\cos(x)$ ning qiymatlari (12 ta pozitsiyadan va ulardan 8 tasi kasr qismi). Ustunlar o‘ng tarafdin tekislanadi.</p> <p>2-formada Oila psixologiyasi fanidan 4 ta variantli test dasturini tuzing.</p>
19.	<p>1. Matnli fayl berilgan. Fayldagi hamma katta lotin harflari kichik harflarga va aksincha, barcha kichik lotin harflari katta harflarga almashtirilsin.</p> <p>2. N butun soni va A, V haqiqiy sonlari berilgan. funksiyasining $[A,B]$ oraliqdagi $(B-A)/N$ qadam bilan hosil bo‘luvchi qiymatlari jadvalini saqlovchi matnli fayl hosil qilinsin. Jadval ikkita ustundan iborat: x argumentli (10 ta pozitsiya va uning 4 tasi kasr qismi) va uning qiymatlari (15 ta pozitsiya va ulardan 8 tasi kasr qismi). Ustunlar o‘ng tarafdin tekislanadi.</p> <p>2-formada Metrologiya fanidan 2 ta variantli test dasturini tuzing.</p>
20.	<p>1. Matnli fayl berilgan. Undagi barcha ketma-ket kelgan probellar bitta probelga almashtirilsin.</p>

	<p>2. Bir xil o'lchamdagi butun sonlardan iborat ikkita fayl berilgan. Shu sonlardan, kengligi 30 ta belgidan iborat bo'lgan ikkita ustunga ajratilgan matnli fayl hosil qilinsin. Birinchi ustunda birinchi boshlang'ich fayl sonlari, ikkinchisida esa ikkinchi boshlang'ich fayl sonlari joylashadi. Matnli fayldagi har bir satrning boshi va oxiriga " " ajratuvchi (kod 124) qo'shilsin. Sonlar ustunning o'ng tarafiga tekislanadi.</p> <p>2-formada Oliy matematika fanidan 3 ta variantli test dasturini tuzing.</p>
21.	<p>1. Uchtadan ko'p bo'lgan satrga ega bo'lgan matnli fayl berilgan. Shu fayldan oxirgi uchta satr o'chirilsin.</p> <p>2. Bir xil o'lchamdagi butun sonlardan iborat ikkita fayl berilgan. Shu sonlardan, kengligi 30 ta belgidan iborat bo'lgan ikkita ustunga ajratilgan matnli fayl hosil qilinsin. Birinchi ustunda birinchi boshlang'ich fayl sonlari, ikkinchisida esa ikkinchi boshlang'ich fayl sonlari joylashadi. Matnli fayldagi har bir satrning boshi va oxiriga " " ajratuvchi (kod 124) qo'shilsin. Sonlar ustunning o'ng tarafiga tekislanadi.</p> <p>2-formada Biznesni boshqarish asoslari fanidan 4 ta variantli test dasturini tuzing.</p>
22.	<p>1. K ($0 < K < 10$) butun son va K ta dan ko'p bo'lgan satrga ega matnli fayl berilgan. Shu faylning oxirgi K ta satri o'chirilsin.</p> <p>2. K (> 25) butun soni va chap tarafdin tekislangan matnli fayl berilgan. Abzas qizil satr (Text26 masalaga qarang) orqali ajratiladi, bo'sh satrlar esa mavjud emas. Matnni shunday formatlangki, uning kengligi K ta belgidan oshmasin va abzaslarga bo'linganligini saqlagan holda chap tarafga tekislanadi. Satrning oxiridagi probellar o'chirilsin. Formatlangan matn yangi faylda saqlansin.</p> <p>2-formada Ma'lumotlar tuzilmasi fanidan 2 ta variantli test dasturini tuzing.</p>
23.	<p>1. K ($0 < K < 10$) butun son va K ta dan ko'p bo'lgan satrga ega matnli fayl berilgan. Boshlang'ich faylning oxirgi K ta elementidan iborat bo'lgan yangi matnli fayl hosil qilinsin.</p> <p>2. K (> 25) butun soni va chap tarafdin tekislangan matnli fayl berilgan. Matn abzasi bitta bo'sh satr orqali ajratiladi. Matnni shunday formatlangki, uning kengligi K ta belgidan oshmasin va abzaslarga bo'linganligini saqlagan holda chap tarafga tekislanadi.</p> <p>2-formada XFX fanidan 3 ta variantli test dasturini tuzing.</p>
24.	<p>1. Matnli fayl berilgan. Agar abzas bitta yoki bir nechta bo'sh satrlar bilan ajratilgan bo'lsa, u holda matndagi abzaslar soni aniqlansin.</p> <p>2. Chap tomonidan tekislangan matnga ega bo'lgan matnli fayl berilgan. Abzaslar bitta bo'sh satr orqali ajratiladi. Har bir bo'sh bo'lmagan satrdagi eng oxirgi probeldan boshlab satr so'zlari orasidagi probellarni qo'shish orqali matn kenglik bo'yicha tekislanadi (ham chap, ham o'ng tomondan tekislanadi). Matn kengligi 50 ta belgiga teng deb olinsin.</p> <p>2-formada Telekommunikatsiya fanidan 4 ta variantli test dasturini tuzing.</p>
25.	<p>1. K butun son va matnli fayl berilgan. Fayldan K-nomerdagi abzas o'chirilsin (abzas bir-biridan bitta yoki bir nechta bo'sh satrlar bilan ajratiladi). O'chirilgan</p>

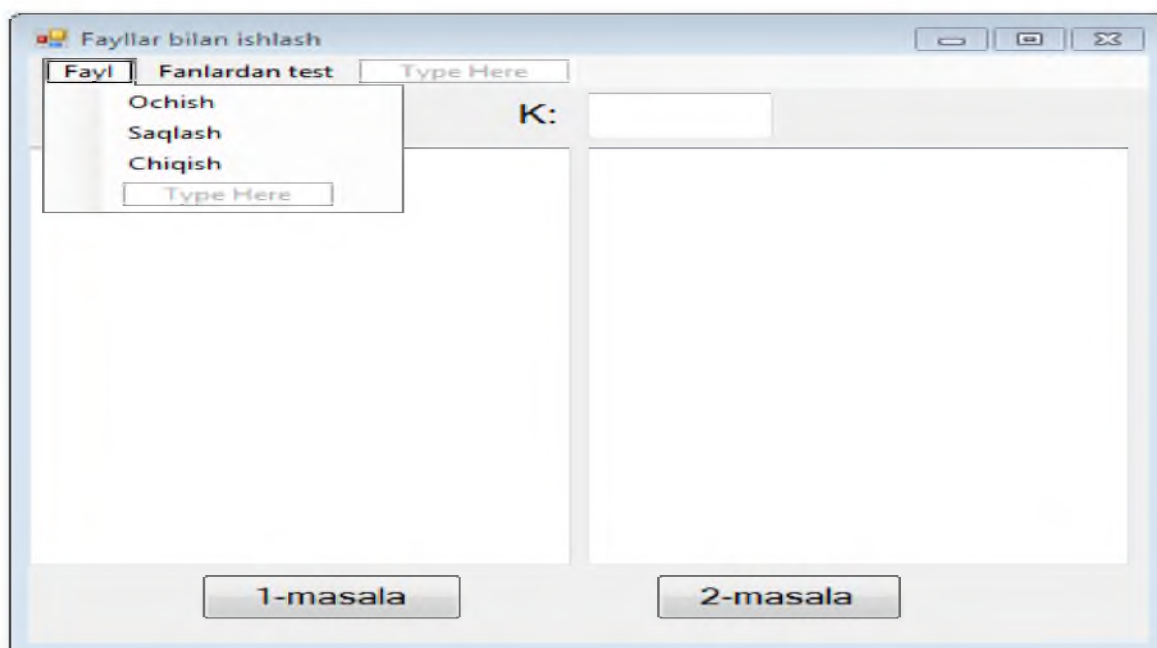
	<p>abzasdan oldin va keyin keluvchi bo'sh satrlar o'chirilmasin. Agar berilgan nomerdagi abzas mavjud bo'lmasa u holda fayl o'zgartirishsiz qoldirilsin.</p> <p>2. O'ng tomonidan tekislangan matnga ega bo'lgan matnli fayl berilgan. Har bir bo'sh bo'lmagan satrning boshlang'ich probellarini yarmini o'chirish orqali matn markazga tekislantirilsin. Toq uzunlikka ega bo'lgan satrlarga, markazlashtirishdan oldin chap tomondan bitta probel o'chirilsin.</p> <p>2-formada Aloqa linyalari fanidan 2 ta variantli test dasturini tuzing.</p>
26.	<p>1. Matnli fayl berilgan. Agar fayl matnidagi har bir abzasning birinchi satri 5 ta probeldan boshlansa ("qizil satr"), u holda shu matndagi abzaslar soni aniqlansin. Abzaslar orasidagi bo'sh satrlar hisobga olinmasin.</p> <p>2. Chap tomonidan tekislangan matnga ega bo'lgan matnli fayl berilgan. Har bir bo'sh bo'lmagan satr boshiga kerakli miqdorda probel qo'shish orqali matnni markazga tekislanilsin. (Matn kengligi 50 ta belgiga teng deb olinsin). Toq uzunlikka ega bo'lgan satrlarga, markazlashtirishdan oldin chap tomondan probel qo'shilsin.</p> <p>2-formada Kompyuter arxitekturasi fanidan 3 ta variantli test dasturini tuzing.</p>
27.	<p>1. Matnli fayl berilgan. Fayl matnidagi har bir abzasning birinchi satri 5 ta probeldan boshlangan. K butun soni va mantli fayl berilgan. Fayldan K-nomerdagi abzas o'chirilsin. Abzas qizil satr orqali ajratiladi.</p> <p>2. Chap tomonidan tekislangan matnga ega bo'lgan matnli fayl berilgan. Har bir bo'sh bo'lmagan satr boshiga kerakli miqdorda probel qo'shish orqali matnni o'ng tomondan tekislanilsin. (Matn kengligi 50 ta belgiga teng deb olinsin).</p> <p>2-formada Tizimli dasturlash fanidan 4 ta variantli test dasturini tuzing.</p>
28.	<p>1. Matnli fayl berilgan. Fayl matnidagi har bir abzasning birinchi satri 5 ta probeldan boshlangan. Abzas <i>qizil satr</i> yordamida ajratiladi. Faylda bo'sh satrlar mavjud emas. Har bir qo'shni abzaslar orasiga bittadan bo'sh satr joylashtirilsin. (faylning boshiga va oxiriga bo'sh satr qo'shilmasin).</p> <p>2. Matnli fayl va lotin harflarining kichik harflaridan S belgi berilgan. Matnli fayl hosil qilinsin va unga boshlang'ich fayldagi barcha shu S harfidan boshlanuvchi so'zlar yozilsin (katta yoki kichik harflar bilan boshlanuvchi). So'z deb, probellarga, tinish belgilariga ega bo'lmagan va probellar, tinish belgilari yoki satrning boshi/oxiri bilan chegaralangan belgilar to'plamiga aytiladi. Agar boshlang'ich fayl tarkibida mos so'zlar mavjud bo'lmasa, u holda natijaviy fayl bo'sh holda qoldirilsin.</p> <p>2-formada Kompyuter tarmoqlari fanidan 3 ta variantli test dasturini tuzing.</p>
29.	<p>1. Matnli fayl berilgan. Matnning eng uzun bo'lgan birinchi so'zi chop qilinsin. So'z deb, probellar bilan chegaralangan yoki satrning boshi/oxiri bo'lgan belgilar to'plamiga aytiladi.</p> <p>2. Matnli fayl va lotin harflarining bosh harflaridan S belgi berilgan. Matnli fayl hosil qilinsin va unga boshlang'ich fayldagi barcha shu S harfidan boshlanuvchi so'zlar yozilsin (katta yoki kichik harflar bilan boshlanuvchi). So'z deb, probellarga, tinish belgilariga ega bo'lmagan va probellar, tinish belgilari yoki satrning boshi/oxiri bilan chegaralangan belgilar to'plamiga aytiladi. Agar</p>

boshlang'ich fayl tarkibida mos so'zlar mavjud bo'lmasa, u holda natijaviy fayl bo'sh holda qoldirilsin.
2-formada Veb texnologiyalar fanidan 2 ta variantli test dasturini tuzing.

14- jadvaldagi topshiriqlarni bajarish uchun uslubiy qo'llanmalar.

Visual C++ dasturida 30 – variantdagi masalani Windows Application muhitida ishlanishi:

1- qadam. Formaga ma'lumot kiritish va chiqarish uchun 2 ta **TextBox1**, **TextBox2**, **K** sonini kiritish uchun **TextBox3**, 1- masalani yechish uchun **Button1**, 2- masala uchun esa **Button2**, ma'lumotlarni ixtiyoriy nom bilan saqlash uchun **openFileDialog1**, **saveFileDialog1** va **menuStrip1** komponentalari joylashtiriladi. Dastur ko'rinishi quyidagicha:



5.4- rasm. Dasturning umumiy ko'rinishi

2- qadam. Button1 ning click hodisasiga quyidagi kodlar yoziladi:

```
1. textBox2->Text="";  
2. char satr[200]=""; int a[100], k=1,l=0; a[l]=0;  
3. for(int i=0; i<textBox1->Text->Length; i++)  
4. { satr[i]=textBox1->Text[i]; }  
5. strcat(satr, " "); int m=0,min=0, max=0;
```

```

6. bool b=true;
7. for(int i=0; i<strlen(satr); i++) {
8. if(satr[i]==‘ ‘){
9. a[l]=k-1; if(b){max=a[l]; b=false;}
10. if(a[l]<=max){max=a[l]; m=i-k; min=k;} l++; k=0;
11. } k++; }
12. char natija[100]="";
13. strncpy(natija,satr+m,min);
14. String^ s=gcnew String(natija);
15.textBox2->Text=s;

```

3- qadam. Button2 ning click hodisasiga quyidagi kodlar yoziladi:

```

1. int K;
2. K=Convert::ToInt64(textBox3->Text);
3. textBox2->Text="";
4. char satr[200]=""; int a[100], k=1,p=1,l=0; a[l]=0;
5. for(int i=0; i<textBox1->Text->Length; i++)
6. { satr[i]=textBox1->Text[i]; }
7. char natija[100]="";
8. strcat(satr," "); int m=0,min=0, max=0;
9. for(int i=0; i<strlen(satr); i++) {
10. if(satr[i]==‘ ‘){
11. if(k-1==K ){
12. strcat(natija,satr+(i-k),k); strcat(natija," "); }
13. k=0; } k++; }
14. String^ s=gcnew String(natija); textBox2->Text=s;

```

4- qadam. 5.1- rasmdagi File menyusidagi Ochish bo‘limining onclick hodisasiga quyidagi kodlar yoziladi:

```

1. openFileDialog1->ShowDialog();
2. if (openFileDialog1->FileName == nullptr) return;
3. try
4. { auto Chitatel = gcnew
5. IO::StreamReader(openFileDialog1-
>FileName,System::Text::Encoding::GetEncoding(1251));
6. textBox1->Text = Chitatel->ReadToEnd();
7. Chitatel->Close();
8. }
9. catch (IO::FileNotFoundException^ Situatsiya)
10. {

```

```

11. MessageBox::Show(Situatsiya->Message + "\n bundey fayl
    yo`q", "Hatolik", MessageBoxButtons::OK,
    MessageBoxIcon::Exclamation);
12. }
13. catch (Exception^ Situatsiya)
14. {
15. MessageBox::Show(Situatsiya->Message, "Hatolik",
    MessageBoxButtons::OK,
16. MessageBoxIcon::Exclamation);
17.}

```

5- qadam. 5.4- rasmdagi File menyusidagi Yopish bo'limining onclick hodisasiga quyidagi kodlar yoziladi:

```

saveFileDialog1->FileName = openFileDialog1->FileName;
if (saveFileDialog1->ShowDialog() ==Windows::Forms::
DialogResult::OK) Zapis();

```

bu yerda Zapis() funksiyasi chaqirilgan. Zapis() funksiyasi quyidagicha:

```

1. void Zapis()
2. {
3. try
4. {
5. auto Pisatel = gcnewIO::StreamWriter(saveFileDialog1-
    >FileName, false,
6. System::Text::Encoding::GetEncoding(1251));
7. Pisatel->Write(textBox1->Text);
8. Pisatel->Close(); textBox1->Modified = false;
9. }
10. catch (Exception^ Situatsiya)
11. {
12. MessageBox::Show(Situatsiya->Message, "Hatolik",
    MessageBoxButtons::OK,
13. MessageBoxIcon::Exclamation);}
14. Chiqish tugmasi kodi quyidacha:
15. private: System::Void
    chiqishToolStripMenuItem_Click(System::Object^ sender,
    System::EventArgs^ e) {
16. this->Close();
    }

```

6- qadam. Formaning Load hodisasining kodlari quyidagicha:

```

1. private: System::Void Form1_Load(System::Object^
    sender, System::EventArgs^ e) {
2. textBox1->Multiline = true; textBox1->Clear();
3. textBox1->Size = Drawing::Size(230, 299);
4. this->Text = "Oddiy matn muharriri"; openFileDialog1-
    >FileName = "C:\\\\Text2.txt";
5. openFileDialog1->Filter = "Текстовые файлы
    (*.txt)|*.txt|All files (*.*)|*.*";
6. saveFileDialog1->Filter = "Текстовые файлы
    (*.txt)|*.txt|All files (*.*)|*.*";}

```

7- qadam. Formaning FormClosing hodisasining kodlari quyidagicha:

```

1. private: System::Void Form1_FormClosing(System::Object^
    sender, System::Windows::Forms::FormClosingEventArgs^
    e) {
2. if (textBox1->Modified == false) return;
3. auto MBox = MessageBox::Show("Текст был изменен.
    \nСохранит изменениya?", "Простой редактор",
    MessageBoxButtons::YesNoCancel, MessageBoxIcon::
    Exclamation);
4. if (MBox == Windows::Forms::DialogResult::No) return;
5. if (MBox == Windows::Forms::DialogResult::Cancel) e-
    >Cancel = true;
6. if (MBox == Windows::Forms::DialogResult::Yes)
7. {
8. if (saveFileDialog1->ShowDialog()
    ==Windows::Forms::DialogResult::OK)
9. { Zapis(); return;
10. }else e->Cancel = true; }

```

7- qadamdan keyin, talabalarni har xil fanlardan baholash uchun yangi forma yaratiladi.

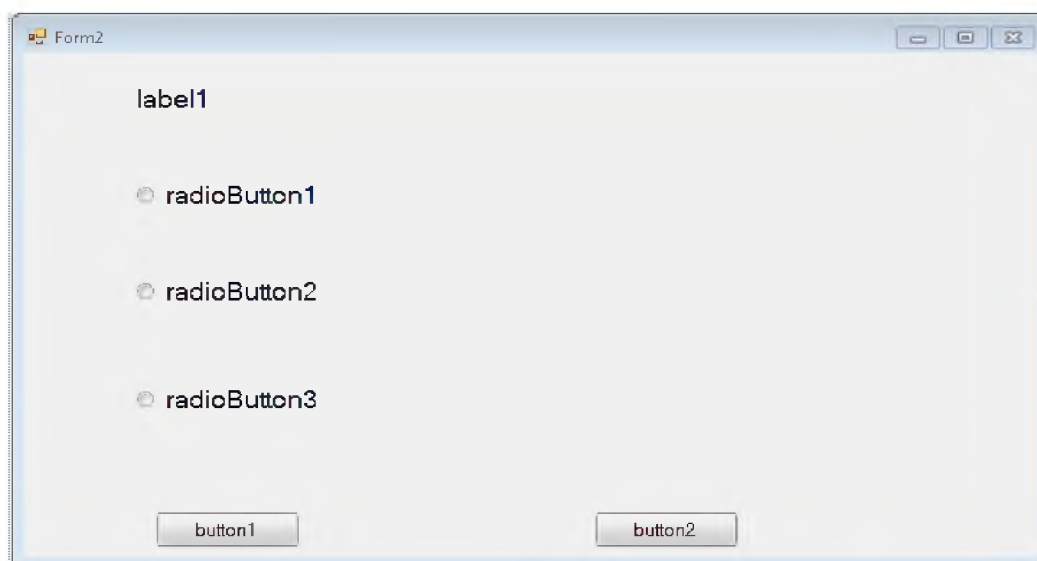
8- qadam. Fanlardat test menyusidagi Informatika bo‘limining onclick hodisasining kodlari quyidagicha:

```

private: System::Void
informatikaToolStripMenuItem_Click(System::Object^ sender,
System::EventArgs^ e) {
    Form2^ op=gcnew Form2();
    this->Hide();
    op->Show(); }

```

9- qadam. 2- formada label1, radioButton1, radioButton2, radioButton3, button1, button2 komponentalari joylashtiriladi. Unig ko‘rinishi quyidagicha:



5.5- rasm. 2- formaning umumiy ko‘rinishi

10- qadam. 2- formadagi **Button1** ning **click** hodisasidagi kodlar quyidagicha:

```
1. private: System::Void button1_Click(System::Object^
  sender, System::EventArgs^ e) {
2. if (Tanlangan_javob == TogriJavobNomeri) TogriJavob =
  TogriJavob + 1;
3. if (Tanlangan_javob != TogriJavobNomeri){
4. NoTogriJavob = NoTogriJavob + 1;
  NoTogri_javoblar[NoTogriJavob] = label1->Text; }
5. if (button1->Text == "Testni boshidan boshlash"){
6. button1->Text = "Keyingi savol";
7. radioButton1->Visible = true; radioButton2->Visible =
  true;
8. radioButton3->Visible = true; Testni_boshalsh(); return;
9. }
10. if (button1->Text == "Yakunlash"){ oqish->Close();
11. radioButton1->Visible = false; radioButton2->Visible =
12. false; radioButton3->Visible = false;
13. label1->Text = String::Format("Test yakunlandi.\n" +
  "Togri javoblar: {0} iz {1}.\n" +
```

```

14. "5 balli tizimda baholash: {2:F2}.", TogriJavob,
    SavollarSoni, (TogriJavob * 5.0F) / SavollarSoni);
    button1->Text = "Testni boshidan boshlash";
15. String^ Str = "Siz javob bergan noto`g`ri savollar
    ro`yhati :\n\n";
16. for (int i = 1; i <= NoTogriJavob; i++)
17. Str = Str + NoTogri_javoblar[i] + "\n";
18. if (NoTogriJavob != 0)
19. MessageBox::Show(Str, "Test yakunlandi");
20. }
21. if (button1->Text == "Keyingi savol")
    Key_savol_oqish();
22. }

```

11- qadam. 2- formaning Formload hodisasi va unda chaqirilgan bir nechta funksiyalarning kodlari quyidagicha:

```

1. #pragma endregion
2. int SavollarSoni;
3. int TogriJavob; int NoTogriJavob; array<String^>^
    NoTogri_javoblar;
4. int TogriJavobNomeri; int Tanlangan_javob;
5. IO::StreamReader^ oqish;
6. private: System::Void Form2_Load(System::Object^
    sender, System::EventArgs^ e) { button1->Text =
    "Keyingi savol";
7. button2->Text = "Chiqish";
8. radioButton1->CheckedChanged +=gcnew EventHandler(this,
    &Form2::Keyingi_savolga_otish);
9. radioButton2->CheckedChanged +=gcnew EventHandler(this,
    &Form2::Keyingi_savolga_otish);
10. radioButton3->CheckedChanged +=gcnew
    EventHandler(this, &Form2::Keyingi_savolga_otish);
11. Testni_boshalsh();
12. void Testni_boshalsh(){System::Text::Encoding^
    Kodirovka = System::Text::Encoding::GetEncoding(1251);
13. try{
14. oqish = gcnew IO::StreamReader(IO::Directory::
    GetCurrentDirectory()+ "\\test.txt", Kodirovka);
15. this->Text = oqish->ReadLine();
16. SavollarSoni = 0; TogriJavob = 0; NoTogriJavob = 0;
17. NoTogri_javoblar = gcnew array<String^>(100);}

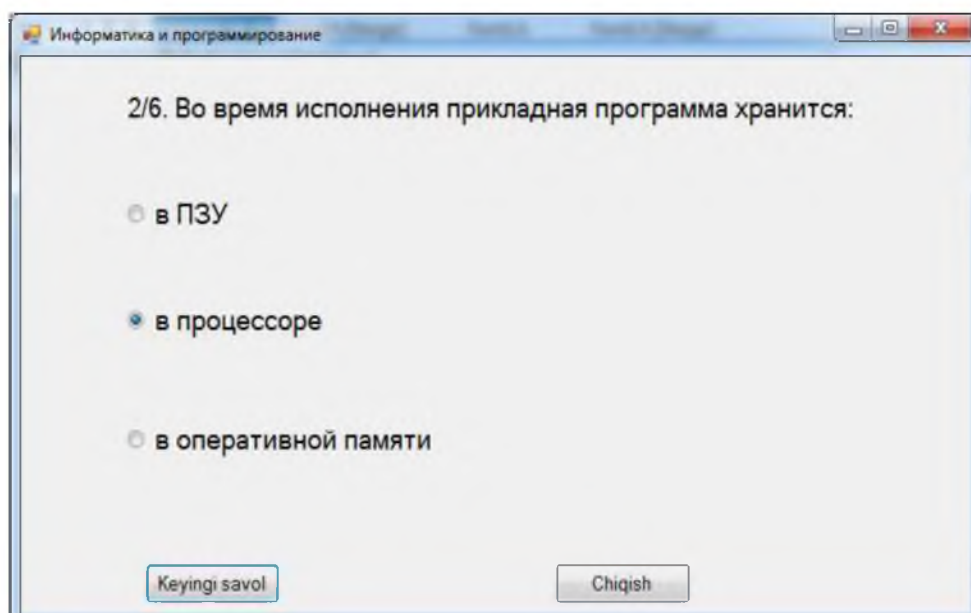
```

```

18. catch (Exception^ Situatsiya)
19. { MessageBox::Show(Situatsiya->Message, "Hatolik",
    MessageBoxButtons::OK, MessageBoxIcon::Exclamation); }
20. Key_savol_oqish(); }
21. void Key_savol_oqish(){
22. label1->Text = oqish->ReadLine();
23. radioButton1->Text = oqish->ReadLine();
24. radioButton2->Text = oqish->ReadLine();
25. radioButton3->Text = oqish->ReadLine();
26. TogriJavobNomeri = int::Parse(oqish->ReadLine());
27. radioButton1->Checked = false; radioButton2->Checked =
    false;
28. radioButton3->Checked = false;
29. button1->Enabled = false;
30. SavollarSoni = SavollarSoni + 1;
31. if (oqish->EndOfStream == true) button1->Text =
    "Zavershit";}
32. private: Void Keyingi_savolga_otish(System::Object^
    sender, System::EventArgs^ e){
33. button1->Enabled = true; button1->Focus();
34. RadioButton^ Pereklyuchatel = (RadioButton^)sender;
    String^ tmp = Pereklyuchatel->Name;
35. Tanlangan_javob = int::Parse(tmp->Substring(11)); }

```

Talabalarni fanlardan baholash dasturini ishga tushirgandan keyingi ko‘rinishi quyidagicha:



5.6- rasm. 2- formaning ishga tushirgandan keyingi ko‘rinishi

7.3. Jadvalli ma'lumotlar asosida va Chart komponentasi yordamida grafik diagrammalar yaratish

Ishning maksadi

TextBox va **dataGridView** komponentalaridan foydalanib, jadvalli ma'lumotlar tiplari yordamida turli diagrammalar yaratuvchi vizual dastur yaratish.

Jadval tipidagi ma'lumotlar yaratish uchun **DataTable** tipidan foydalaniladi. Bu tipda yaratilgan o'zgaruvchi ustun va satrlardan tashkil topadi. Jadval ko'rinishidagi ma'lumotlar asosida grafik diagrammalar qurish oson bo'ladi. Formaning grafik imkoniyatlarini o'zida jamlagan komponenta bu **Chart** komponentasidir. Ushbu komponenta yordamida jadvalli ma'lumotlarni ekranga grafik diagrammalar ko'rinishida tasvirlash mumkin. Ushbu laboratoriya ishini mazmunini har oyda sotib olingan mahsulotlar miqdorlarini jadvallarda shakillantirib, grafik diagramma ko'rinishida tasvirlovchi vizual dastur yaratish orqali tushintirib o'tamiz. Topshiriqdagi variantingiz masalasining dasturini tuzishda ushbu dastur namuna sifatida hizmat qiladi. Lekin diagrammalarni ko'rinishlari va turlari har xil bo'lganligi uchun ushbu turlardan ham foydalanib tuzilsa dasturimizning tashqi ko'rinishlari chiroyli va ko'rgazmali bo'ladi.

Topshiriq

Ma'ruza va amaliyot darslarida olingan ma'lumotlarga asoslanib, o'z variantingiz uchun quyidagilarni bajaring:

- a) 15- jadvaldan o'z variantini ko'chirib oling;
- b) quyidagi namuna asosida o'z variantingizdagi masalalarni dasturini tuzing.

Har-bir variantning 1- formasida 2 ta masalani yechish dasturini tuzing.

15- jadval

No	Masala sharti
1	1-formada ishlab chiqarish do'koniga kiruvchi maxsulotlarni xajmi, ularga to'lanadigan pul miqdorini haftalik diagrammasini va 2-formada oylar kesimidagi diagrammasini yaratuvchi vizual dastur tuzing.

2	1-formada ishlab chiqarish do‘konining ishchilariga to‘lanadigan usta xaqlarni summasini oylar kesimidagi diagrammasini va 2-formada esa sof daromatni oylar kesimidagi diagrammasini yaratuvchi vizual dastur tuzing.
3	1-formada kompyuter qurilmalarining narxlari kesimidagi diagrammasini va 2-formada esa sifat kesimidagi diagrammasini yaratuvchi vizual dastur tuzing.
4	1-formada guruhdagi talabalarining fanlardan olgan baholari kesimidagi diagrammasini va 2-formada esa fanlarga ajratilgan soatlar kesimidagi diagrammasini yaratuvchi vizual dastur tuzing.
5	1-formada Antivirus dasturlarining narxlari kesimidagi diagrammasini va 2-formada esa sifat kesimidagi diagrammasini yaratuvchi vizual dastur tuzing.
6	1-formada Kopyuter qurilmalarini sotuvchi do‘konlarning narxlari kesimidagi diagrammasini va 2-formada esa xajmi kesimidagi diagrammasini yaratuvchi vizual dastur tuzing.
7	1-formada NoteBook larning parametrlari kesimidagi diagrammasini va 2-formada esa NoteBook ishlab chiqaruvchi firmalarning ishlab chiqarish ko‘rsatgichlari bo‘yicha diagrammasini yaratuvchi vizual dastur tuzing.
8	1-formada Guruh talabalarining yoshlari bo‘yicha diagrammasini va 2-formada esa C++da dasturlash fanidan olgan baholari bo‘yicha diagrammasini yaratuvchi vizual dastur tuzing.
9	1-formada Ob-havoning namlik miqdorini hafta kunlari bo‘yicha diagrammasini va 2-formada esa issiqlik darajalari bo‘yicha diagrammasini yaratuvchi vizual dastur tuzing.
10	1-formada Ishlab chiqarish fabrikasida tikuv mashinalariga sarflanadigan xarajatlarni oylar bo‘yicha diagrammasini va 2-formada esa tushumni va sof foydani hafta kunlari bo‘yicha diagrammasini yaratuvchi vizual dastur tuzing.
11	1-formada AQSh valyuta kursini bir haftalik ko‘rsatgichlarini sumga nisbatan diagrammasini va 2-formada esa rublga nisbatan diagrammasini yaratuvchi vizual dastur tuzing.
12	1-formada Bozorda guruch narxini oylar bo‘yicha diagrammasini va 2-formada esa shakar narxlarini oylar bo‘yicha diagrammasini yaratuvchi vizual dastur tuzing.
13	1-formada Supermarketda xaridorlar kelishini hafta kunlari bo‘yicha diagrammasini va 2-formada esa sotilgan maxsulotlarni xajmi bo‘yicha oylarga nisbatan diagrammasini yaratuvchi vizual dastur tuzing.
14	1-formada Toshkentdagi kinoteatrlarga keluvchi tomoshabinlar sonini hafta kunlari bo‘yicha diagrammasini va 2-formada esa qaysi kinolarga ko‘proq qiziqishlar bo‘layotganligini aniqlovchi diagrammasini yaratuvchi vizual dastur tuzing.
15	1-formada Toshkentdagi serqatnov ostanovkalarda qaysi raqamdagi avtobuslarga talablar ko‘proq bo‘layotganligini aniqlovchi diagrammasini va 2-formada esa qaysi raqamdagi avtobuslar oraliq masofasi eng kam bo‘layotganlarini kunlik diagrammasini yaratuvchi vizual dastur tuzing.

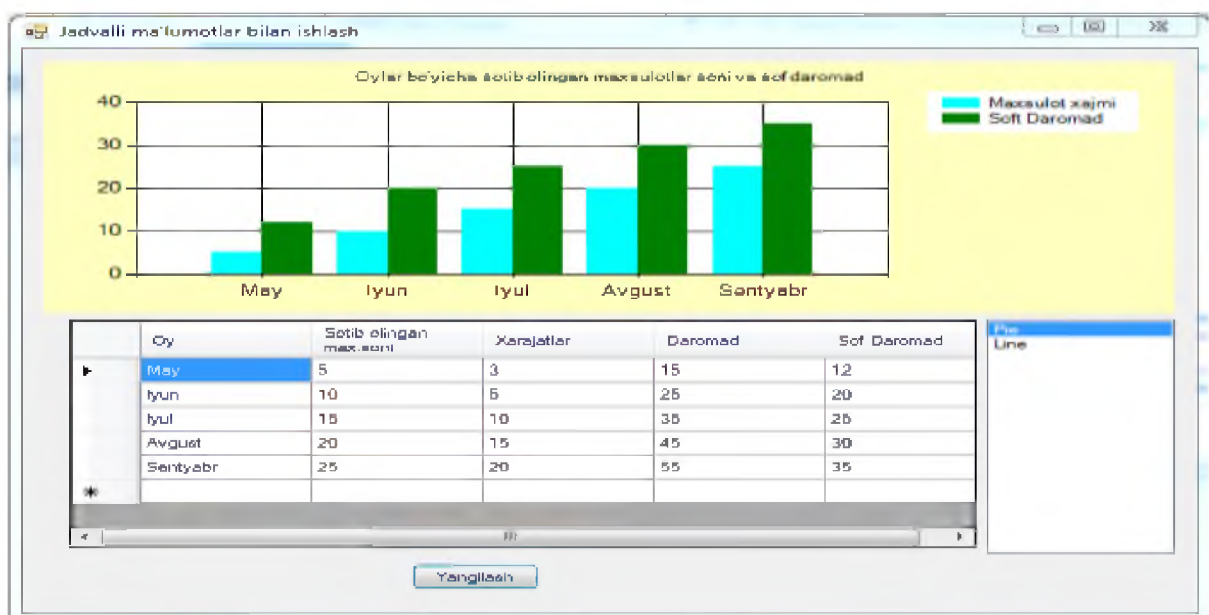
16	1-formada O'zbekistonda qaysi oynai jaxon telekanallarini ko'proq tamosha qilinayotglnligini aniqlovchi diagrammasini va 2-formada esa qaysi teledasturlarga tomoshabinlardan qiziqish ko'p bo'layotganlarini anqlovchi oylik diagrammasini yaratuvchi vizual dastur tuzing.
17	1-formada TATUning fakultetlaridagi talabalar soni bo'yicha ko'rsatgichlar diagrammasini va 2-formada esa abuturentlardan qaysi fakultetlarga ko'proq qiziqishlar bo'layotganligini bo'yicha diagrammasini yaratuvchi vizual dastur tuzing.
18	1-formada TATUda C++dasturlash tiliga bo'lgan qiziqishlarni fakultetlar bo'yicha diagrammasini va 2-formada esa C++ da dasturlash fanidan talabalarni reyting ko'rsatgichlarini yo'nalishlar bo'yicha diagrammasini yaratuvchi vizual dastur tuzing.
19	1-formada Interaktiv xizmatlar ko'rsatuvchi veb saytlarga bo'lgan qiziqishlarni oylar bo'yicha diagrammasini va 2-formada esa elektron pochta qutilaridan foydalanish ko'rsatkichlari bo'yicha diagramma yaratuvchi vizual dastur tuzing.
20	1-formada Toshkentda oylik yo'l chiptalarini sotilishi xajmini oylar bo'yicha diagrammasini va 2-formada esa oylik yo'l chiptalarinining qaysi turiga ko'proq talab bo'layotganlik ko'rsatkichlari bo'yicha diagramma yaratuvchi vizual dastur tuzing.
21	1-formada TATU kutubxonalarida qanday turdagi kitoblarga bo'lgan qiziqishlar diagrammasini va 2-formada esa qaysi kutbxona bo'limlariga bo'layotganlik qiziqishlari bo'yicha diagramma yaratuvchi vizual dastur tuzing.
22	1-formada Universitet bitiruvchilarining sifat ko'rsatkichlarini yillarga nisbatan diagrammasini va 2-formada esa qaysi yo'nalishdagi bituruvchilarga bo'lgan qiziqishlarni firmalar bo'yicha diagrammasini yaratuvchi vizual dastur tuzing.
23	1-formada Shakar narxining o'zgarishlarini oylar kesidagi diagrammasini va 2-formada esa oziq yog'i narxining oylarga nisbatan ko'tarilishi va pasayishi ko'rsatgichlarini oylar bo'yicha diagrammasini yaratuvchi vizual dastur tuzing.
24	1-formada do'konda CocaCola ichimligidan tushadigan sof foydani kunlik diagrammasini va 2-formada esa do'konlardan undiriladigan soliqlarni topiladigan sof daromatga nisbatan diagrammasini yaratuvchi vizual dastur tuzing.
25	1-formada Guruh talabalarini ON va JN natijalari diagrammasini va 2-formada esa YaN diagrammasini yaratuvchi vizual dastur tuzing.
26	1-formada Bozorlarda go'sht va guruch maxsulotlarining narxlarni oylar bo'yicha o'sish ko'rsatgichlari diagrammasini va 2-formada esa sut maxsulotlarni oylar bo'yicha diagrammasini yaratuvchi vizual dastur tuzing.
27	1-formada Zargarlik do'konida tillaning narxini oylar bo'yicha o'sish ko'rsatgichlari diagrammasini va 2-formada esa kumushnining narxini oylar bo'yicha diagrammasini yaratuvchi vizual dastur tuzing.

28	1-formada Toshkentda oylik yo‘l chiptalarining sotilish sonini oylar bo‘yicha diagrammasini va 2-formada esa internet kartochkalarini sotilishi xajmini oylar bo‘yicha diagrammasini yaratuvchi vizual dastur tuzing.
29	1-formada Ishlab chiqarish korxonasida maxsulot ishlab chiqarish uchun sarflanadigan xarajatlarni haftalik diagrammasini va 2-formada esa sof foydani xajmini oylar bo‘yicha diagrammasini yaratuvchi vizual dastur tuzing.
30	1-formada Firmada har oyda sotib olingan mahsulotlar miqdorlarini va sof daromatini oylik diagrammasini yaratuvchi vizual dastur tuzing.

15- jadvaldagi topshiriqlarni bajarish uchun uslubiy qo‘llanmalar.

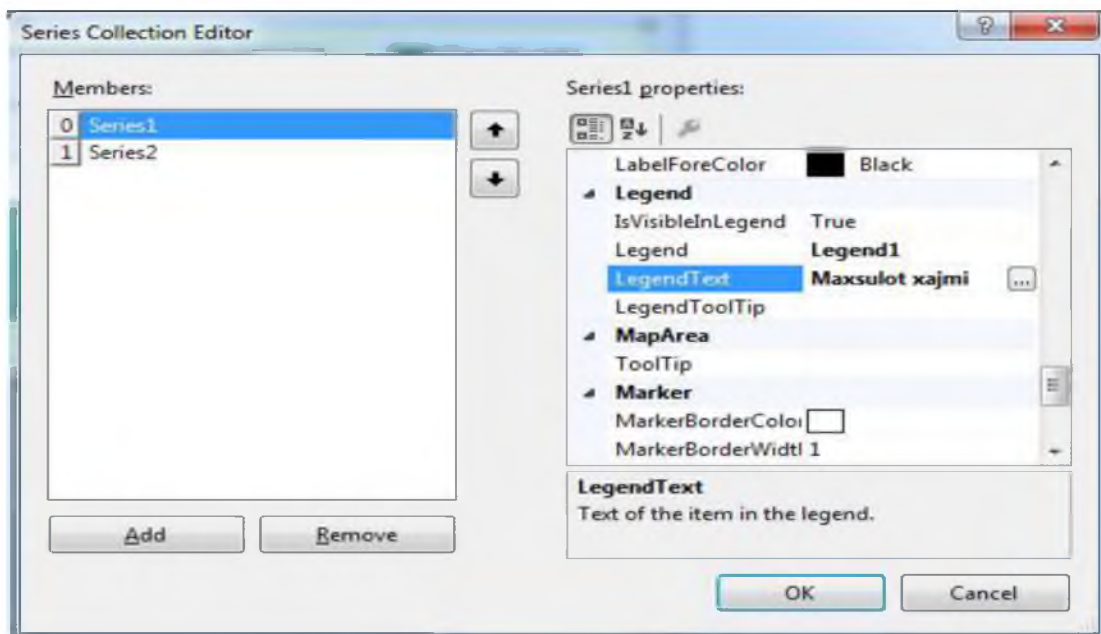
Visual C++ dasturida 30 – variantdagi masalani Windows Application muhitida ishlanishi:

Dasturning umumiy ko‘rinishi quyidagicha bo‘ladi.



7.7- rasm. Dasturning umumiy ko‘rinishi oynasi

1- qadam. Formaga **chart** komponentasi joylashtiriladi. Uning **Series** xossasiga kirib, 2- **Series** qo‘shiladi. **Series1** tanlanadi va uning **LegendText** xossasiga “**Maxsulot hajmi**” matni yoziladi. **Series2** tanlanadi va uning **LegendText** xossasiga “**Soft daromat**” matni yoziladi. Ushbu **Series** qo‘shish oynasi **5.8-** rasmda keltirilgan.



7.8- rasm. Series qo‘shish oynasi

2- qadam. Formaga `dataGridView`, `listBox` va `button` komponentalari joylashtiriladi. Ularning hech qanday xosasi o‘zgartirilmaydi.

3- qadam. Formaning `load` hodissasining kodlari quyidagicha:

1. `DataTable ^ Tablitsa;`
2. `private: System::Void Form1_Load(System::Object^ sender, System::EventArgs^ e) {chart1->Titles->Add("Oylar bo‘yicha sotib olingan maxsulotlar soni va sof daromat");`
3. `listBox1->Items->Add("Pie");`
4. `listBox1->Items->Add("Line");`
5. `this->Text = "Grafik ma‘lumotlar";`
6. `Tablitsa = gcnew DataTable();`
7. `//Ushbu jadvalda 2 ta ustun "Oy"va "Sotib olingan //max.soni":`
8. `Tablitsa->Columns->Add("Oy", String::typeid);`
9. `// V C#: Tablitsa.Columns.Add("Mesyats", //typeof(String));`
10. `//2-ustun bo‘yicha ma‘lumotlar long tipida bo‘lasi:`
11. `Tablitsa->Columns->Add("Sotib olingan max.soni", Int64::typeid); Tablitsa->Columns->Add("Xarajatlar", Int64::typeid);`
12. `Tablitsa->Columns->Add("Daromat", Int64::typeid); Tablitsa->Columns->Add("Sof daromat", Int64::typeid);`

```

13. // V C#: Tablitsa.Columns.Add("Obʻem prodaj",
    typeof(long));
14. //Jadvalning 1- qatorini toʻldiramiz:
15. DataRow ^ Ryad = Tablitsa->NewRow();
16. Ryad["Oy"] = "May"; Ryad["Sotib olingan max.soni"]
    =10; Ryad["Xarajatlar"] =3; Ryad["Daromat"] =15;
17. Ryad["Sof daromat"]
    =Convert.ToInt64(Ryad["Daromat"])-
    Convert.ToInt64(Ryad["Xarajatlar"]);
18. //Convert.ToInt64( dataGridView1->Rows[0]->Cells[0]-
    >Value);
19. Tablitsa->Rows->Add(Ryad);
20. //Jadvalning 2- qatorini toʻldiramiz:
21. Ryad = Tablitsa->NewRow();
22. Ryad["Oy"] = "Iyun"; Ryad["Sotib olingan max.soni"] =
    20; Ryad["Xarajatlar"] =5; Ryad["Daromat"] =25;
23. Ryad["Sof daromat"]
    =Convert.ToInt64(Ryad["Daromat"])-
    Convert.ToInt64(Ryad["Xarajatlar"]);
24. Tablitsa->Rows->Add(Ryad);
25. //3- qatorni qoʻshamiz:
26. Ryad = Tablitsa->NewRow();
27. Ryad["Oy"] = "Iyul"; Ryad["Sotib olingan max.soni"] =
    30; Ryad["Xarajatlar"] =10; Ryad["Daromat"] =35;
28. Ryad["Sof daromat"]
    =Convert.ToInt64(Ryad["Daromat"])-
    Convert.ToInt64(Ryad["Xarajatlar"]);
29. Tablitsa->Rows->Add(Ryad);
30. //4- qatorni qoʻshamiz:
31. Ryad = Tablitsa->NewRow();
32. Ryad["Oy"] = "Avgust"; Ryad["Sotib olingan max.soni"]
    = 40; Ryad["Xarajatlar"] =15; Ryad["Daromat"] =45;
33. Ryad["Sof daromat"]
    =Convert.ToInt64(Ryad["Daromat"])-
    Convert.ToInt64(Ryad["Xarajatlar"]);
34. Tablitsa->Rows->Add(Ryad);
35. //5- qatorni qoʻshamiz:
36. Ryad = Tablitsa->NewRow();
37. Ryad["Oy"] = "Sentyabr"; Ryad["Sotib olingan
    max.soni"] = 50; Ryad["Xarajatlar"] =20; Ryad["Daromat"]
    =55;

```

```

38. Ryad["Sof daromat"]
    =Convert::ToInt64(Ryad["Daromat"])-
    Convert::ToInt64(Ryad["Xarajatlar"]);
39. Tablitsa->Rows->Add(Ryad);
40. dataGridView1->DataSource = Tablitsa;
41. chart1->DataSource = Tablitsa;
42. chart1->Series["Series1"]->XValueMember = "Oy";
43. chart1->Series["Series2"]->XValueMember = "Oy";
44. chart1->Series["Series1"]->YValueMembers = "Sotib
    olingan max.soni";
45. chart1->Series["Series2"]->YValueMembers = "Sof
    daromat";
46. chart1->Series["Series1"]->ChartType =
    System::Windows::Forms::DataVisualization::Charting::Ser
    iesChartType::Column;
47. chart1->Series["Series2"]->ChartType =
    System::Windows::Forms::DataVisualization::Charting::Ser
    iesChartType::Column;
48. // Diogrammaning turi quyidagicha bo'lishi ham
    //mumkin. masalan: Pie, Line va bohqa.
49. chart1->Series["Series1"]->Color = Color::Aqua;
    chart1->Series["Series2"]->Color = Color::Yellow;
50. chart1->DataBind();}
51. Button1 tugmasini click xodissasining kodlari quyidagicha:
52. private: System::Void button1_Click(System::Object^
    sender, System::EventArgs^ e) {
53. dio();
54. }
55. void dio(){
56. this->Text = "Grafik ma'lumotlar";
57. Tablitsa = gnew DataTable();
58. Tablitsa->Columns->Add("Oy", String::typeid);
59. Tablitsa->Columns->Add("Sotib olingan max.soni",
    Int64::typeid); Tablitsa->Columns->Add("Xarajatlar",
    Int64::typeid);
60. Tablitsa->Columns->Add("Daromat", Int64::typeid);
    Tablitsa->Columns->Add("Sof daromat", Int64::typeid);
61. DataRow ^ Ryad = Tablitsa->NewRow();
62. Ryad["Oy"] = dataGridView1->Rows[0]->Cells[0]->Value;
    Ryad["Sotib olingan max.soni"] =dataGridView1->Rows[0]-
    >Cells[1]->Value;

```

```

63. Ryad["Xarajatlar"] =dataGridView1->Rows[0]->Cells[2]-
    >Value; Ryad["Daromat"] =dataGridView1->Rows[0]-
    >Cells[3]->Value;
64. Ryad["Sof daromat"] =Convert::ToInt64(dataGridView1-
    >Rows[0]->Cells[3]->Value) -
    Convert::ToInt64(dataGridView1->Rows[0]->Cells[2]-
    >Value);
65. Tablitsa->Rows->Add(Ryad);
66. Ryad = Tablitsa->NewRow();
67. Ryad["Oy"] = dataGridView1->Rows[1]->Cells[0]->Value;
    Ryad["Sotib olingan max.soni"] = dataGridView1->Rows[1]-
    >Cells[1]->Value;
68. Ryad["Xarajatlar"] =dataGridView1->Rows[1]->Cells[2]-
    >Value; Ryad["Daromat"] =dataGridView1->Rows[1]-
    >Cells[3]->Value;
69. Ryad["Sof daromat"] =Convert::ToInt64(dataGridView1-
    >Rows[1]->Cells[3]->Value) -
    Convert::ToInt64(dataGridView1->Rows[1]->Cells[2]-
    >Value);
70. Tablitsa->Rows->Add(Ryad);
71. Ryad = Tablitsa->NewRow();
72. Ryad["Oy"] = dataGridView1->Rows[2]->Cells[0]->Value;
    Ryad["Sotib olingan max.soni"] = dataGridView1->Rows[2]-
    >Cells[1]->Value;
73. Ryad["Xarajatlar"] =dataGridView1->Rows[2]->Cells[2]-
    >Value; Ryad["Daromat"] =dataGridView1->Rows[2]-
    >Cells[3]->Value;
74. Ryad["Sof daromat"] =Convert::ToInt64(dataGridView1-
    >Rows[2]->Cells[3]->Value) -
    Convert::ToInt64(dataGridView1->Rows[2]->Cells[2]-
    >Value);
75. Tablitsa->Rows->Add(Ryad);
76. Ryad = Tablitsa->NewRow();
77. Ryad["Oy"] = dataGridView1->Rows[3]->Cells[0]->Value;
    Ryad["Sotib olingan max.soni"] = dataGridView1->Rows[3]-
    >Cells[1]->Value;
78. Ryad["Xarajatlar"] =dataGridView1->Rows[3]->Cells[2]-
    >Value; Ryad["Daromat"] =dataGridView1->Rows[3]-
    >Cells[3]->Value;
79. Ryad["Sof daromat"] =Convert::ToInt64(dataGridView1-
    >Rows[3]->Cells[3]->Value) -

```



```

    Convert::ToInt64(dataGridView1->Rows[3]->Cells[2]-
    >Value);
80. Tablitsa->Rows->Add(Ryad);
81. Ryad = Tablitsa->NewRow();
82. Ryad["Oy"] = dataGridView1->Rows[4]->Cells[0]->Value;
    Ryad["Sotib olingan max.soni"] = dataGridView1->Rows[4]-
    >Cells[1]->Value;
83. Ryad["Xarajatlar"] =dataGridView1->Rows[4]->Cells[2]-
    >Value; Ryad["Daromat"] =dataGridView1->Rows[4]-
    >Cells[3]->Value;
84. Ryad["Sof daromat"] =Convert::ToInt64(dataGridView1-
    >Rows[4]->Cells[3]->Value) -
    Convert::ToInt64(dataGridView1->Rows[4]->Cells[2]-
    >Value);
85. Tablitsa->Rows->Add(Ryad);
86. chart1->DataSource = Tablitsa;
87. chart1->Series["Series1"]->XValueMember = "Oy";
88. chart1->Series["Series2"]->XValueMember = "Oy";

89. chart1->Series["Series1"]->YValueMembers = "Sotib
    olingan max.soni";
90. chart1->Series["Series2"]->YValueMembers = "Sof
    daromat";

91. chart1->Series["Series1"]->ChartType =
    System::Windows::Forms::DataVisualization::Charting::Ser
    iesChartType::Column;
92. chart1->Series["Series2"]->ChartType =
    System::Windows::Forms::DataVisualization::Charting::Ser
    iesChartType::Column;
93. chart1->Series["Series1"]->Color = Color::Aqua;
94. chart1->Series["Series2"]->Color = Color::Green;
95. //chart1->Series["Series1"]->IsVisibleInLegend =
    false;
96. chart1->DataBind();
97. dataGridView1->DataSource = Tablitsa;
98.}

```

listBox1 komponentasining **SelectedIndexChanged** hodissasini kodlari quyidagicha:


```

1. private: System::Void
   listBox1_SelectedIndexChanged(System::Object^ sender,
   System::EventArgs^ e)
2. {
3. if (listBox1->Text == "Transparent") return;
4. this->chart1->Series["Series1"]->ChartTypeName
   =(listBox1->Text) ;
5. this->chart1->Series["Series2"]->ChartTypeName
   =(listBox1->Text) ;
6. }

```

4- qadam. Jadvaldan kiritilgan qiymatlar natijaga darhol tasir qilishi uchun `dataGridView1` ning `CellEndEdit` hodissasiga `dio()` funksiyasi chaqiriladi. Ahamiyat bergan bo'lsa, `button1` ga ham chaqirilgan edi. Ushbu kod quyidagicha:

```

private: System::Void
dataGridView1_CellEndEdit(System::Object^ sender,
System::Windows::Forms::DataGridViewCellEventArgs^ e)
{
    dio();
}

```

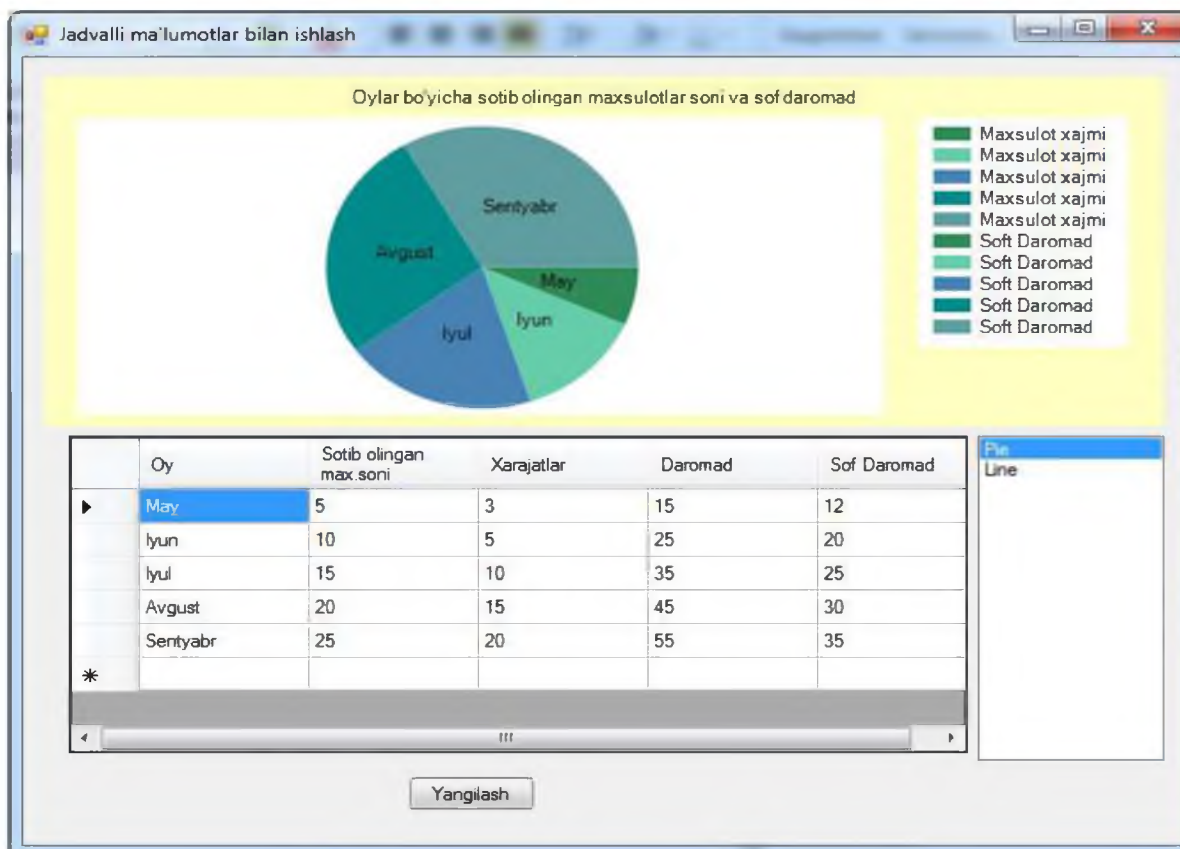
5- qadam. `listBox1`da diagrammani ko'rinishlarini o'zgartirish mumkin. Buning uchun `listBox` ning `SelectedIndexChanged` hodisasining kodlari quyidagicha bo'lishi kerak:

```

1. private: System::Void
   listBox1_SelectedIndexChanged(System::Object^ sender,
   System::EventArgs^ e)
2. {
3. if (listBox1->Text == "Transparent") return;
4. this->chart1->Series["Series1"]->ChartTypeName
   =(listBox1->Text) ;
5. this->chart1->Series["Series2"]->ChartTypeName
   =(listBox1->Text) ;
6. }

```

Dasturda diagrammani ko'rinishi quyidagicha:



7.9- rasm. Diagrammani sozlash oynasi

7.4. Web brauzerda HTML jadvallarni tasvirlash va shakllantirish

Ishning maksadi

WebBrowser komponentasidan foydalanib, matnli faylda joylashgan ma'lumotlarni ekranga jadval ko'rinishida chiqaruvchi vizual dasturlar yaratish.

Bizga ma'lumki HTML teglari yordamida Veb ilovalar yaratiladi va bu ilova jamlanib veb sayt bo'ladi. Veb ilovalarni yoki global tarmoqqa ulangan saytlarni amaliy dasturda ochish imkonini beruvchi komponenta bu WebBrowser komponentasi hisoblanadi. Ushbu komponenta orqali nafaqat saytlarni ochish balki HTML kodlarida terilgan tekst(teg)larni ham chiqarish va tahrirlash imkoniyatlari mavjud. Quyidagi na'munada tuziladigan dasturda HTML teglari ishlatilgan. Masalan: <h1>-matnни 16 shiriftda ko'rsatadi. <table>- jadval yaratadi. <tr>- jadvalning ustunini yaratadi. <td>- jadvalning satrlarini yaratadi. 8- jadvalda

keltirilgan ma'lumotlar matnli faylda tashkil qilinadi. Ushbu fayl o'qish uchun ochiladi va uning ma'lumotlari ma'lum o'zgaruvchilarga o'zlashtirilgan holda, sikl asnosida webBrowser komponentasi yordamida ekranga tartiblangan ko'rinishida chiqariladi.

Topshiriq

Ma'ruza va amaliyot darslarida olingan ma'lumotlarga asoslanib, o'z variantingiz uchun quyidagilarni bajaring:

- a) 16- jadvaldan o'z variantingizni ko'chirib oling,
- b) quyidagi namuna asosida o'z variantingizdagi masalalarning dasturini tizing.

16- jadval

№	<u>Topshiriq nomi</u>	<u>Matnli fayldagi ma'lumotlar</u>
1	Inson	Familiyasi, ismi, jinsi, millati, e'tiqod qiladigan dini, bo'yi vazni, tug'ilgan sanasi (yil, oy, kun), telefon raqami, uy manzili (pochta indeksi, mamlakat, viloyat, tuman, shahar, ko'cha, uy, xonadon).
2	Maktab o'quvchisi	Familiyasi, ismi, otasining ismi, sinf nomeri, jinsi, tug'ilgan sanasi (yil, oy, kun), uy manzili, sinfi raxbari, bahosi.
3	Xaridor	Familiyasi, ismi, otasining ismi, manzili, tuman, shahar, ko'cha, uy raqami, xonadon raqami, kredit kartochkasi raqami yoki hisob raqami.
4	Bemor	Familiyasi, ismi, tug'ilgan yili, telefon raqami, tumani, uy manzili, tibbiy kartochka raqami, qon guruhi.
5	Sport komandasi	Nomi, shahri, nechta o'yin o'ynaganligi, qancha ochko to'plaganligi (yutqazgan, yutgan, durang), o'yinchilar soni.
6	Stadion	Nomi, sport turi, qurilgan sanasi, manzili, tomoshabinlar sig'imi, arenalar, maydonchalar soni, sektor raqami.
7	Avtomobil egasi	Familiyasi, ismi, avtomobil raqami, avtomobil nomi, texpasport raqami, tug'ilgan yili, telefon raqami, qayd qilingan YPX bo'limi.
8	Avtomobil	Markasi, rangi, seriyasi nomeri, qayd raqami, eshiklari soni, chiqqan yili, narxi.
9	Film	Nomi, rejissori (familiyasi, ismi), mamlakati, ekranga chiqqan yili, narxi (ketgan xarajatlar), olingan daromad.

10	Musiqiy mahsulot	Tashuvchi (gramplastinka, audiokasseta, lazerli disk), katalogdagi tartib raqami, nomi, ijrochisi (famiyasi, ismi), ijro davomiyligi, asarlar soni, katalog bo'yicha narxi.
11	Vokal-cholg'u guruhi albomi	Guruhning nomi, albomning nomi, kasseta, diskdagi qo'shiqlar soni, albom chiqqan yil, ishlab chiqaruvchi firma.
12	Davlat	Mamlakat nomi, poytaxti, davlat tili, aholisi (soni), yer maydoni, pul birligi valyutasining so'm (dollar)ga nisbatan kursi, davlat tuzumi, qit'asi.
13	Monitor	Davlat, ishlab chiqarilgan yili, markasi, narxi, kafolat vaqti, o'lchami, rangi, netto va boshqalar.
14	Kitob	Muallifi, ham muallif, asar nomi, beti, yili, sohasi, janri, narxi va boshqalar.
15	Antivirus	Nomi, versiyasi, narxi, kalit vaqti, firmasi va boshalar.
16	Telefon	Nomi, markasi, yili, o'lchami, narxi, kamera o'lchami, operasion tizim tipi, garantiya, davlati va boshqalar.
17	Qaydnoma	1-joriy nazorat (jn), 1-joriy nazorat qayta ishlash (jnqi), 1-oraliq nazorat (on), 1-oraliq nqi, 2-on, 2-onqi, jamon OT + ON, yakuniy nazorat, o'zlashtirish ko'rsatkichi, reyting bali.
18	O'yinchoq	Nomi, narxi, davlati, bolalar yoshiga chegara va boshqalar.
19	Talaba	Famiyasi, ismi, sharifi, tug'ilgan sanasi (kun oy yil), guruhi, bahosi va boshqalar.
20	Yugurish	Famiyaya, ismi, sharifi, davlati, yoshi, masofa, shaxsiy rekordi vaqti vaqti, o'rni.
21	Printer	Nomi, markasi, davlati, ishlab chiqarilgan yili, 1 minutda chiqaradigan varaqlar soni, o'g'irligi, kartrij raqami, toneri, rangi, og'irligi va boshqalar.
22	RAM	Nomi, o'lchami, ishlab chiqarilgan yili, narxi, davlati va boshqalar.
23	Futbolchi	Famiyasi, ismi, sharifi, tug'ilgan sanasi (kun oy yil), fuqaroligi, ampulasi, klubi, gollar soni va boshqalar.
24	Qog'oz	Markasi, o'lchami, narxi, og'irligi, rangi, davlati va boshqalar.
25	Shkaf	Nomi, turi, rangi, material, o'lchami, narxi, davlati va boshqalar.
26	Daftar	Nomi, varaqlar soni, qog'ozi, firmasi, abloshkasi (qattiq, silliq), rangi, o'lchami va boshqalar
27	Universitet	Nomi, tashkil topgan yili, manzili (viloyati, tumani, ko'chasi, raqami), telefoni, faksi, talabalar soni va boshqalar.

28	Bino	Nomi, manzili, qurilgan yili, qurgan tashkilot, ajratilgan mablag‘, yer maydoni, bino maydoni, etajlar soni, qurish usuli va boshqalar.
29	Mahalla	Nomi, tumani, shirkati, aholi soni, oqsoqoli, telefoni, ko‘p qavatli uylar soni, oilalar soni va boshqalar.
30	Mashina	Nomi, dvigatel nomi, sig‘imi, ot kuchi, rangi, narxi, davrlari va boshqalar.
31	Oilam haqidagi ma‘lumotlar	Qarindoshlik turi, Familiyasi, Ismi, Tel nomeri,

16- jadvaldagi topshiriqlarni bajarish uchun uslubiy qo‘llanmalar.

Visual C++ dasturida 31 – variantdagi masalani Windows Application muhitida ishlanishi:

Dasturning umumiy ko‘rinishi quyidagicha bo‘ladi:



7.10- rasm. Dasturning umumiy ko‘rinish oynasi

1- qadam. Formaga **webBrowser** komponentasini tashlang va **Form1**ning **load** hodissasining kodlari quyidagicha:

1. **#pragma endregion**
2. **private: System::Void Form1_Load(System::Object^ sender, System::EventArgs^ e)**
3. {

```

4. String^s="<h1 align=\"center\">Oilam haqi
   ma`lumot<br><table border=\"1\"> ";
5. s+="<td>Nomeri<td>";
6. s+="<td>Qar_turi<td>";
7. s+="<td>Fam<td>";
8. s+="<td>Ismi<td>";
9. s+="<td>Tel nomeri<td>";
10. char qar_turi[20]="";
11. char Ismi[20]="";
12. char Fam[20]="";
13. char Tel[20]="";
14. String^ss="";
15. FILE *oqi;
16. int i=1,n;
17. bool dd;
18. oqi=fopen("C:\\1.txt","r");
19. if(oqi==NULL){MessageBox::Show("Not found");
20. }
21. while(!feof(oqi))
22. {
23. fgets(qar_turi,20,oqi); fgets(Fam,20,oqi);
24. fgets(Ismi,20,oqi); dd=fgets(Tel,20,oqi);
25. if(dd==0){
26. break;
27. }
28. String^ Fq=gcnew String(qar_turi);
29. String^ Fi=gcnew String(Ismi);
30. String^ Fm=gcnew String(Fam);
31. String^ Ft=gcnew String(Tel);
32. s+="<tr><td>"+i.ToString()+"<td>";
   s+="<td>"+Fq+"<td>";
33. s+="<td>"+Fm+"<td>";
34. s+="<td>"+Fi+"<td>";
35. s+="<td>"+Ft+"<td>";
36. i++;
37. }
38. s+="</table> ";
39. webBrowser1->Navigate("about:" + s);
40. }

```

2- qadam. Formada saytlarni qidirish uchun **Button** va **TextBox** komponentalari joylashtiriladi va **Button1** tugmasining **click** xodissasi kodlari quyidagicha:

```
private: System::Void button1_Click(System::Object^ sender,
System::EventArgs^ e)
{
    if ( !this->textBox1->Text->Equals( "" ) )
    {
        this->webBrowser1->Navigate( this->textBox1->Text );
    }
}
```

3- qadam. Adreslar oynasiga veb sayt manzili yozilgandagi oyna hodisasi quyidagicha bo'ladi:

```
private: System::Void textBox1_KeyDown(System::Object^
sender, System::Windows::Forms::KeyEventArgs^ e)
{
    if ( e->KeyCode == System::Windows::
Forms::Keys::Enter && !this->textBox1->Text->Equals( "" ) )
    {
        this->webBrowser1->Navigate( this->textBox1->Text );
    }
}
```

4- qadam. **webBrowser1** komponentasining **Navigated** hodisasi kodlari quyidagicha:

```
private: System::Void webBrowser1_Navigated(System:: Object^
sender, System::Windows::Forms::
WebBrowserNavigatedEventArgs^ e)
{
    this->textBox1->Text = this->webBrowser1->Url->ToString();
}
```

7.5. Visual C++da MS word imkoniyatlaridan foydalanib, jadvallar yaratish, ularni word faylga eksport qilish va taqdim etish

Ishning maksadi

Visual C++ muharririda **Microsoft MS Word**ning jadvallar yaratish imkoniyatlaridan foydalanib, **Word** faylda jadvallarni shakillantiruvchi vizual dasturlar yaratish.

Visual C++ da **Microsoft MS Office** paketidagi amaliy dasturlarning imkoniyatlaridan foydalanish imkoniyatlari mavjud. Ushbu qo'llanmaning 2-laboratoriya ishida **Visual Basic** tilidagi **InputBox** kiritish oynasi imkoniyatlaridan foydalanilgan edi. Laboratoriya ishida **Microsoft MS Office** ning jadval yaratish usullari imkoniyatlaridan qanday foydalanish yo'llari o'rganiladi. Ish **Visual Studio** dasturini ishga tushirishdan boshlanadi. **File** menyu bo'limidan **New Project** tanladi va yangi loyiha yaratiladi. Ushbu loyihaga **MS Word** ning ob'ektlar kutubxonasi qo'shialadi. Buning uchun **Project** menyu bo'limidan **Add Reference** burug'i tanlanadi yoki **Alt+F7** tugmalari bosiladi. Ochilgan **Property Pages** oynadan **Com** bo'limiga o'tiladi va **Add new Reference** tugmasi bosiladi. Ochilgan oynadan **Microsoft Word 14.0 Object Library 8.5** belgilanadi va **OK** tugmachasi bosiladi. So'ngra ushbu kutubxona muavffaqiyatli qo'shilganligi haqida ma'lumot chiqadi. Natijada dasturda ushbu kutubxona funksiyalaridan foydalanish imkoniyati vujudga keladi.

Topshiriq

Ma'ruza va amaliyot darslarida olingan ma'lumotlarga asoslanib, o'z variantingiz uchun quyidagilarni bajaring:

- a) 17- jadvaldagi topshiriqlar asosida jadvallar va **String** tipidagi massivlar tashkil qiling va bu massivlar bilan **Microsoft Word** kutubxonasidagi jadval yaratish funksiyasi yordamida jadval yarating;
- b) jadvallarni **Microsoft Word** dasturida ochuvchi vuzual dastur tizing.

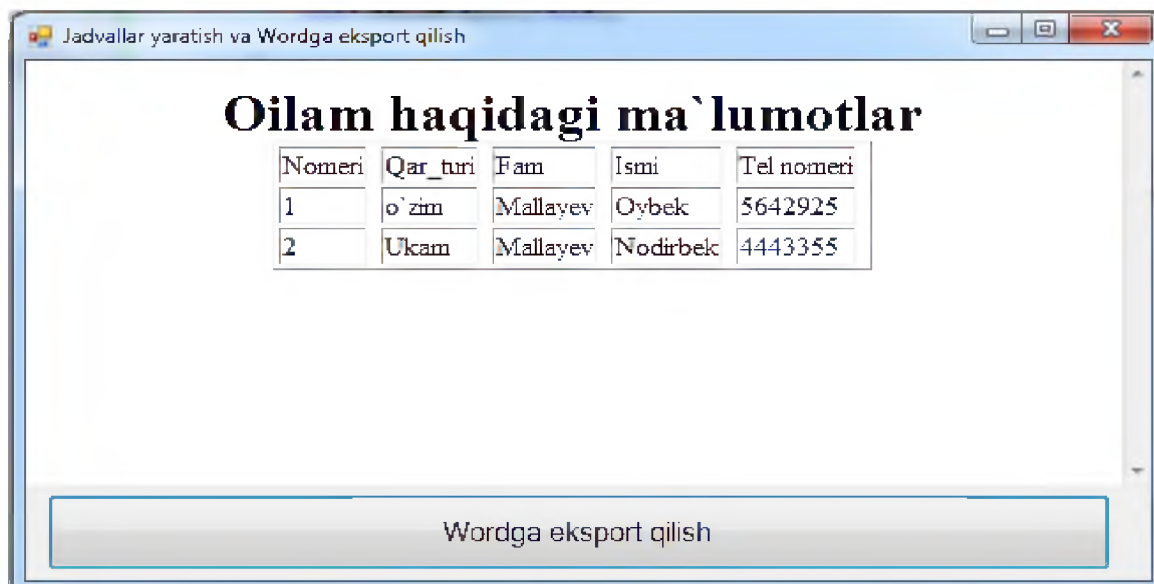
№	Jadval nomi	Jadval ustunlarining nomlari
1	Inson	Familiyasi, ismi, jinsi, millati, e'tiqod qiladigan dini, bo'yi vazni, tug'ilgan sanasi (yil, oy, kun), telefon raqami, uy manzili (pochta indeksi, mamlakat, viloyat, tuman, shahar, ko'cha, uy, xonadon).
2	Maktab o'quvchisi	Familiyasi, ismi, otasining ismi, sinf nomeri, jinsi, tug'ilgan sanasi (yil, oy, kun), uy manzili, sinfi raxbari, bahosi.
3	Xaridor	Familiyasi, ismi, otasining ismi, manzili, tuman, shahar, ko'cha, uy raqami, xonadon raqami, kredit kartochkasi raqami yoki hisob raqami.
4	Bemor	Familiyasi, ismi, tug'ilgan yili, telefon raqami, tumani, uy manzili, tibbiy kartochka raqami, qon guruhi.
5	Sport komandasi	Nomi, shahri, nechta o'yin o'ynaganligi, qancha ochko to'plaganligi (yutqazgan, yutgan, durang), o'yinchilar soni.
6	Stadion	Nomi, sport turi, qurilgan sanasi, manzili, tomoshabinlar sig'imi, arenalar, maydonchalar soni, sektor raqami.
7	Avtomobil egasi	Familiyasi, ismi, avtomobil raqami, avtomobil nomi, texpasport raqami, tug'ilgan yili, telefon raqami, qayd qilingan YPX bo'limi.
8	Avtomobil	Markasi, rangi, seriyasi nomeri, qayd raqami, eshiklari soni, chiqqan yili, narxi.
9	Film	Nomi, rejissori (familiyasi, ismi), mamlakati, ekranga chiqqan yili, narxi (ketgan xarajatlar), olingan daromad.
10	Musiqiy mahsulot	Tashuvchi (gramplastinka, audiokasseta, lazerli disk), katalogdagi tartib raqami, nomi, ijrochisi (familiyasi, ismi), ijro davomiyligi, asarlar soni, katalog bo'yicha narxi.
11	Vokal-cholg'u guruhi albomi	Guruhning nomi, albomning nomi, kasseta, diskdagi qo'shiqlar soni, albom chiqqan yil, ishlab chiqaruvchi firma.
12	Davlat	Mamlakat nomi, poytaxti, davlat tili, aholisi (soni), yer maydoni, pul birligi valyutasining so'm (dollar)ga nisbatan kursi, davlat tuzumi, qit'asi.
13	Monitor	Davlat, ishlab chiqarilgan yili, markasi, narxi, kafolat vaqti, o'lchami, rangi, netto va boshqalar.
14	Kitob	Muallifi, ham muallif, asar nomi, beti, yili, sohasi, janri, narxi va boshqalar.
15	Antivirus	Nomi, versiyasi, narxi, kalit vaqti, firmasi va boshqalar.

16	Telefon	Nomi, markasi, yili, o'lchami, narxi, kamera o'lchami, operasion tizim tipi, garantiya, davlati va boshqalar.
17	Qaydnom a	1-joriy nazorat (jn), 1-joriy nazorat qayta ishlash (jnqi), 1-oraliq nazorat (on), 1-oraliq nqi, 2-on, 2-onqi, jamon OT + ON, yakuniy nazorat, o'zlashtirish ko'rsatkichi, reyting bali.
18	O'yinchoq	Nomi, narxi, davlati, bolalar yoshiga chegara va boshqalar.
19	Talaba	Familiyasi, ismi, sharifi, tug'ilgan sanasi (kun oy yil), guruhi, bahosi va boshqalar.
20	Yugurish	Familiya, ismi, sharifi, davlati, yoshi, masofa, shaxsiy rekordi vaqti vaqti, o'rni.
21	Printer	Nomi, markasi, davlati, ishlab chiqarilgan yili, 1 minutda chiqaradigan varaqlar soni, o'g'irligi, kartrij raqami, toneri, rangi, og'irligi va boshqalar.
22	RAM	Nomi, o'lchami, ishlab chiqarilgan yili, narxi, davlati va boshqalar.
23	Futbolchi	Familiyasi, ismi, sharifi, tug'ilgan sanasi (kun oy yil), fuqaroligi, ampulasi, klubi, gollar soni va boshqalar.
24	Qog'oz	Markasi, o'lchami, narxi, og'irligi, rangi, davlati va boshqalar.
25	Shkaf	Nomi, turi, rangi, material, o'lchami, narxi, davlati va boshqalar.
26	Daftar	Nomi, varaqlar soni, qog'ozi, firmasi, abloshkasi (qattiq, silliq), rangi, o'lchami va boshqalar
27	Univer-sitet	Nomi, tashkil topgan yili, manzili (viloyati, tumani, ko'chasi, raqami), telefoni, faksi, talabalar soni va boshqalar.
28	Bino	Nomi, manzili, qurilgan yili, qurgan tashkilot, ajratilgan mablag', yer maydoni, bino maydoni, etajlar soni, qurish usuli va boshqalar.
29	Mahalla	Nomi, tumani, shirkati, aholi soni, oqsoqoli, telefoni, ko'p qavatli uylar soni, oilalar soni va boshqalar.
30	Mashina	Nomi, dvigatel nomi, sig'imi, ot kuchi, rangi, narxi, davrlari va boshqalar.
31	Oilam haqidagi ma'lumotlar	Qarindoshlik turi, Familiyasi, Ismi, Tel nomeri,

17- jadvaldagi topshiriqlarni bajarish uchun uslubiy qo'llanmalar.

Visual C++ dasturida 31 – variantdagi masalani Windows Application muhitida ishlanishi:

Dasturning umumiy ko'rinishi quyidagicha bo'ladi:



5.11- rasm. Dasturning umumiy koʻrinishi oynasi

1- qadam. Form1ga asosiy dastur oynasida matnli fayl maʼlumotlarini chiqarish uchun **webBrowser** komponentasini tashlang va **Form1** ning **load** hodissasi kodlari quyidagicha boʻladi:

```

1. #pragma endregion
2. private: System::Void Form1_Load(System::Object^ sender,
   System::EventArgs^ e)
3. {
4. button2->Text = "Wordga eksport qilish";
5. this->Text = "Jadvallar yaratish va Wordga eksport
   qilish";
6. String^s="<h1 align=\"center\">Oilam haqidagi
   ma'lumotlar<br><table border=\"1\"> ";
7. s+="<td>Nomeri<td>";
8. s+="<td>Qar_turi<td>";
9. s+="<td>Fam<td>";
10. s+="<td>Ismi<td>";
11. s+="<td>Tel nomeri<td>";
12. char qar_turi[20]="";
13. char Ismi[20]="";
14. char Fam[20]="";
15. char Tel[20]="";
16. String^ss="";
    // fayllar bilan ishlash oqimi

```

```

    //fayldagi ma'lumotlarni o'zlashtirish uchun 4 ta char
    //turidagi massiv e'lon qilish
17. FILE *oqi;
18. int i=1,n;
19. bool dd;
20. oqi=fopen("C:\\word.txt","r");
21. // faylga yo'l ko'rsatish
22. if(oqi==NULL){MessageBox::Show("Fayl topilmadi");}
23. while(!feof(oqi)){
24. // Matnli fayldan ma'lumotlarni o'zlashtirish
25. fgets(qar_turi,20,oqi);
26. fgets(Fam,20,oqi);
27. fgets(Ismi,20,oqi);
28. dd=fgets(Tel,20,oqi);
29. if(dd==0){break;}
30. // Matnli faylda ohirgi ma'lumotni aniqlash
31. // char turidan String turiga o'tkazish
32. String^ Fq=gcnew String(qar_turi);
33. String^ Fi=gcnew String(Ismi);
34. String^ Fm=gcnew String(Fam);
35. String^ Ft=gcnew String(Tel);
36. // s o'zgaruvchisiga o'zlashtirish
37. s+="<tr><td>"+i.ToString()+"<td>";
38. s+="<td>"+Fq+"<td>";
39. s+="<td>"+Fm+"<td>";
40. s+="<td>"+Fi+"<td>";
41. s+="<td>"+Ft+"<td>";
42. //s+="<td>564-29-25<td>";
43. i++;
44. }
45. s+="</table> ";
46. webBrowser1->Navigate("about:" + s);
47. }

```

2- qadam. Form1ga asosiy dastur oynasida matnli fayl ma'lumotlarini word faylga eksport qilish uchun **Button** komponentasini tashlang va uning **OnClick** xodissasi kodlari quyidagicha bo'ladi:

```

1. private: System::Void button2_Click(System::Object^
    sender, System::EventArgs^ e) {
2. // 4 ta satrli massiv e'lon qilish

```

```

3. array<String^> ^
   Fi={"Ismi", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", ""
   ", "", "", "", "", "", "", "", "", ""};
4. array<String^> ^
   Ff={"Familiyasi", "", "", "", "", "", "", "", "", "", "", "", "", "", "", ""
   ", "", "", "", "", "", "", "", "", ""};
5. array<String^> ^
   Ft={"Telefoni", "", "", "", "", "", "", "", "", "", "", "", "", "", "", ""
   ", "", "", "", "", "", "", "", "", ""};
6. array<String^> ^
   Fq={"Qarin_turi", "", "", "", "", "", "", "", "", "", "", "", "", "", "", ""
   ", "", "", "", "", "", "", "", "", ""};
7. FILE *oqi; // fayllar bilan ishlash oqimi
8. //fayldagi ma'lumotlarni o'zlashtirish uchun 4 //ta char
   turidagi massiv e'lon qilish
9. char qar_turi[20]="";
10. char Ismi[20]="";
11. char Fam[20]="";
12. char Tel2[20]="";
13. int i=1,n;
14. bool dd;
15. oqi=fopen("C:\\word.txt", "r");
16. // faylga yo'l ko'rsatish
17. if(oqi==NULL){MessageBox::Show("Fayl topilmadi");}
18. while(!feof(oqi)){
19. // Matnli fayldan ma'lumotlarni o'zlashtirish
20. fgets(qar_turi,20,oqi);
21. fgets(Fam,20,oqi);
22. fgets(Ismi,20,oqi);
23. dd=fgets(Tel2,20,oqi);
24. if(dd==0){break;}
25. // Matnli faylda ohirgi ma'lumotni aniqlash
26. // char turidan Stting turiga o'tkazish
27. // va massiv elementlariga o'zlashtirish
28. Fq-> SetValue(gcnew String(qar_turi),i);
29. Ff-> SetValue(gcnew String(Fam),i);
30. Fi-> SetValue(gcnew String(Ismi),i);
31. Ft-> SetValue(gcnew String(Tel2),i);
32. i++;
33. }
34. // Word::_Application sinfi nushasini yaratish:
35. auto Vord1 =

```

```

36. gcnew Microsoft::Office::Interop::Word:: Application();
37. Vord1->Visible = true;
38. // "bo'sh" qiymatli o'zgaruvchi:
39. auto t = Type::Missing;
40. // Yaegi MS Word xujjat ochish:
41. auto Dokument = Vord1->Documents->Add(t, t, t, t);
42. // MS WORD xujjatiga joriy xolati bilan matnlar
    kiritish:
43. Vord1->Selection->TypeText("Oilam haqida ma'lumotlar");
44. // Katakchalar parametri va ularni chegaralarini
    //ko'rsatish:
45. System::Object ^ t1 = Microsoft::Office::Interop::
46. Word::WdDefaultTableBehavior::wdWord9TableBehavior;
47. // Parametr, ukazyvayushiy budet li prilojeniye Word
    //avtomaticheski
48. //Jadvaldagi katakchalar o'lchamini o'zgartirish:
49. System::Object ^ t2 = Microsoft::Office::Interop::
50. Word::WdAutoFitBehavior::wdAutoFitContent;
51. // i ta satr va 5 ta ustunli jadval yaratish:
52. Vord1->ActiveDocument->Tables->Add(Vord1->Selection-
    >Range,
53. i, 5, t1, t2); int ii=i;
54. // Jadval katakchalarini to'ldirish:
55. for (int i = 1; i <= ii; i++)
56. {
57. if(i>=2)Vord1->ActiveDocument->Tables[1]->Cell(i, 1)-
    >default->InsertAfter((i-1).ToString());
58. Vord1->ActiveDocument->Tables[1]->Cell(i, 2)->
59. default->InsertAfter(Fq[i - 1]);
60. Vord1->ActiveDocument->Tables[1]->Cell(i, 3)->
61. default->InsertAfter(Ff[i - 1]);
62. Vord1->ActiveDocument->Tables[1]->Cell(i, 4)->
63. default->InsertAfter(Fi[i - 1]);
64. Vord1->ActiveDocument->Tables[1]->Cell(i, 5)->
65. default->InsertAfter(Ft[i - 1]);
66. // C# tilida, quyidagicha yoziladi:
67. // Vord1.ActiveDocument.Tables[1].Cell(i, 2).
68. // Range.InsertAfter(Tel[i - 1]);
69. }
70. // MS Word xujjatlari ilovalarini belgilash:
71. Object ^ t3 =
    Microsoft::Office::Interop::Word::WdUnits::wdLine;

```

```

72. //MS Word xujjatiga i ta satrni aniylash :
73. Object ^ stroka9 = i;
74. Vord1->Selection->MoveDown(t3, stroka9, t);
75. // Quyidagi matnni chop qilish:
76. Vord1->Selection->TypeText("Kafedra mudiri
    Sh.A. Sadullayeva");
77. // Xujjatni avtomatik saqlash, lekin buni foydalanuvchi
    hal qilgani maqul:
78. Object ^ ImyaFayla = "C:\\Natija.doc";
79. Vord1->ActiveDocument->SaveAs(ImyaFayla, t, t, t, t, t,
    t, t, t, t, t, t, t, t, t);
80. }

```

Dastur tahlili:

Ushbu dastur kodida 4 ta **String** tipidagi massiv e'lon qilindi. Dasturning kutubxona e'lon qilish bo'limiga fayllar bilan ishlash direktivasi (**#include<fstream>**) qo'shildi. **Word::Application** sinf ob'ekti yaratildi(**auto Vord1 =gnew Microsoft::Office::Interop::Word:: Application();**) va uning **Add** metodidan foydalanib **Word** fayl ochildi. Matnli fayldagi ma'lumotlarni **FILE** tipidagi ko'rsatgich yordamida 4 ta massivga o'zlashtirildi. Massivdagi ma'lumotlar (**Vord1->ActiveDocument->Tables[1]->Cell(i,2)->default-> InsertAfter(Fq[i - 1]);**) faylga o'zlashtirildi.

7.6. Visual C++ da MS Excell imkoniyatlaridan foydalanib, diagrammalar yaratish va ularni turli kengaytmalarda saqlash

Ishning maksadi

Visual C++ muharririda **Microsoft MS Excell**ning diagrammalar yaratish imkoniyatlaridan foydalanib, **Excell** faylda diagrammalarni shakillantiruvchi vizual dasturlar yaratish.

Visual C++ da **Microsoft MS Office** paketidagi amaliy dasturlarning imkoniyatlaridan foydalanish imkoniyatlari mavjud. Jumladan jadvallar redaktori **MS Excell** da murakkab hisoblashlarni amalga oshirish imkoniyatlari mavjud. Ya'niy dasturlash imkoniyatlarini ham oshiradi. Ushbu imkoniyatlardan qanday foydalanishni ushbu laboratoriya ishida ko'rib chiqiladi. Jumladan **Microsoft MS Excell** ning diagrammalar yaratish usullari imkoniyatlaridan foydalanish yo'llari o'rganiladi. Bazi hisoblashlarni diagrammalar orqali namoish qilish dasturning vizualligini orttiradi. Bu esa har doim **MS Excell** ning ma'lumotlari o'zraganda vizual diagrammani ham o'zgarganini ko'rish mumkin. Ish **Visual Studio** dasturini ishga tushirishdan boshlanadi. **File** menyu bo'limidan **New Project** tanladi va yangi loyiha yaratiladi. Ushbu loyihaga **MS Excell** ning ob'ektlar kutubxonasi qo'shialadi. Buning uchun **Project** menyu bo'limidan **Add Reference** burug'i tanlanadi yoki **Alt+F7** tugmalari bosiladi. Ochilgan **Property Pages** oynadan **Com** bo'limiga o'tiladi va **Add new Reference** tugmasi bosiladi. Ochilgan oynadan **Microsoft Excell 14.0 Object Library 8.5** belgilanadi va **OK** tugmachasi bosiladi. So'ngra ushbu kutubxona muvaffaqiyatli qo'shilganligi haqida ma'lumot chiqadi. Natijada dasturda ushbu kutubxona funksiyalaridan foydalanish imkoniyati vujudga keladi.

Topshiriq

Ma'ruza va amaliyot darslarida olingan ma'lumotlarga asoslanib, o'z variantingiz uchun quyidagilarni bajaring.

- a) 17- jadvaldagi topshiriqlar asosida **Microsoft Excell** imkoniyatlaridan foydalanib Diagrammalar yarating;
- b) diagrammalarni **Microsoft Excell** dasturida ochuvchi vizual dastur tizing.

17- jadvaldagi topshiriqlarni bajarish uchun uslubiy qo'llanmalar.

Visual C++ dasturida 31 – variantdagi masalani Windows Application muhitida ishlanishi:

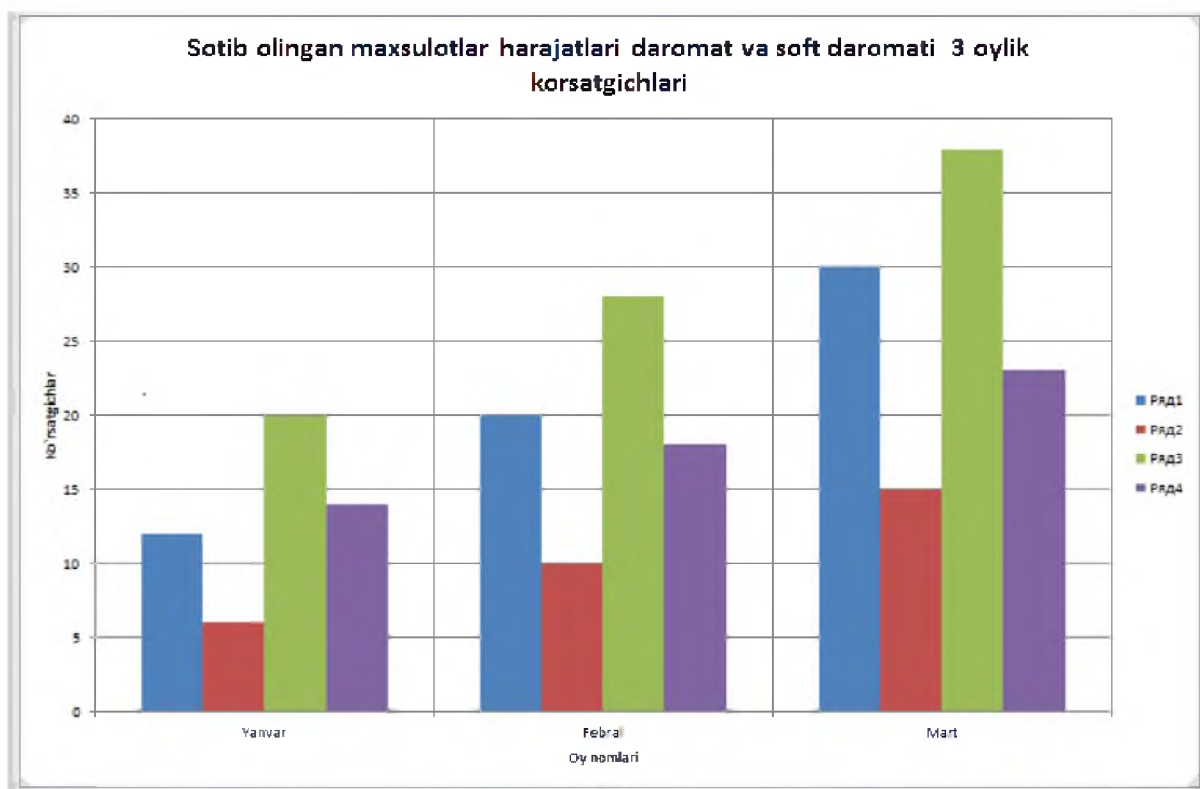
Dasturning umumiy ko'rinishi quyidagicha bo'ladi:


```

C:\Users\Oybek\Documents\Visual Studio 2012\Projects\Metodichka\MS Excell bilan ishlash\MS_E...
Necha oylik daromatni hisoblamogchisiz? 3
Sotib olingan maxsulotlar sonini, harajat va daromatni oylar kesimida kiriting:
Yanvar: Maxsulotlar soni: 12
Harajatlar soni: 6
Daromatlar soni: 20
Febral:
Maxsulotlar soni: 20
Harajatlar soni: 10
Daromatlar soni: 28
Mart:
Maxsulotlar soni: 30
Harajatlar soni: 15
Daromatlar soni: 38

```

5.12- rasm. Dasturning kompilyatsiya vaqtidagi oynasi



5.13- rasm. Dasturning kompilyatsiyadan keyingi diagramma oynasi

1- qadam. Dasturga forma muhiti talab qilinmaganligi uchun loyihani **Console Application** da tuziladi. **Visula C++** dasturi ishga tushiriladi. **File** menyusu

bo‘limidan **New Project** tanladi, **CLR** qismidan **Console Application CLR** muhitida yangi loyiha yaratiladi. MS Excell kutubxonasi qo‘shiladi (**Add Reference->Com->Add new Reference->Microsoft Excell 14.0 Object Library** 8.5). Dastur kodlari quyidagicha bo‘ladi:

```
1. // MS_Excell_bilan_ishlash.cpp : main project file.
2. #include "stdafx.h"
3. #include "iostream";
4. using namespace System;
5. using namespace Microsoft::Office::Interop::Excel;
6. using namespace std;
7. int main(array<System::String ^> ^args)
8. {
9.     double harajat[12], daromat[12], Sof_daromat[12],
        maxsulot_soni[12]; int n;
10.    bir:Console::Write("Necha oylik daromatni
        hisoblamogchisiz?");
11.    std::cin>>n;
12.    Console::WriteLine("Sotib olingan maxsulotlar sonini,
        harajat va daromatni oylar kesimida kiriting:");
13.    if(n<=12){
14.        for(int i=0;i<n;i++){
15.            if(i==0){Console::Write("Yanvar: ");}
16.            else if(i==1){Console::WriteLine("Febral: ");}
17.            else if(i==2){Console::WriteLine("Mart: ");}
18.            else if(i==3){Console::WriteLine("Aprel: ");}
19.            else if(i==4){Console::WriteLine("May: ");}
20.            else if(i==5){Console::WriteLine("Iyun: ");}
21.            else if(i==6){Console::WriteLine("Iyul: ");}
22.            else if(i==7){Console::WriteLine("Avgust: ");}
23.            else if(i==8){Console::WriteLine("Sentyabr: ");}
24.            else if(i==9){Console::WriteLine("Oktyabr: ");}
25.            else if(i==10){Console::WriteLine("Noyabr: ");}
26.            else if(i==11){Console::WriteLine("Dekabr: ");}
27.            Console::Write("Maxsulotlar soni: ");
                std::cin>>maxsulot_soni[i];
28.            Console::Write("Harajatlar soni:
                ");std::cin>>harajat[i];
29.            Console::Write("Daromatlar soni:
                ");std::cin>>daromat[i];
```

```

30. Sof_daromat[i]= daromat[i]-harajat[i];
31. if(i==n-1){
32. // Excel::Application klasidan nusxa olish:
33. _Application ^ XL1 = gcnew Application();
34. XL1->Visible = true;
35. // mavjud metodda foydalanish uchun parametr berish:
36. Object ^ t = Type::Missing;
37. // MS Excel da yangi varoq yaratish:
38. Workbook ^ Kniga = XL1->Workbooks->Add(t);
39. // Varoqni faylda e'lon qilish:
40. Sheets ^ Listy = Kniga->Worksheets;
41. // Birinchi listni tanlash:
42. _Worksheet ^ List = (_Worksheet ^)Listy->Item[1];
43. // Agar yana bitta list qo'shishni kerak bo'lsa quyidagi
//kodni terish lozim:
44. // _Worksheet ^ List = safe_cast<_Worksheet^>(Listy-
>Item[ (Object^)1 ]);

45. //Ma'lumotlarni joriy parametrlari bilan Diagramma
yaratishga buyurtma berish:
46. _Chart ^ Grafik = (_Chart ^)XL1->Charts->Add(t, t, t,
t);
47. if(n>=1){
48. List->Range["A1", t]->Value2 = "Oylar nomi"; List-
>Range["A2", t]->Value2 = "Yanvar";
49. List->Range["B1", t]->Value2 = "Sotib olingan maxsulot
soni"; List->Range["B2", t]->Value2 = maxsulot_soni[0];
50. List->Range["C1", t]->Value2 = "Harajatlar"; List-
>Range["C2", t]->Value2 = harajat[0];
51. List->Range["D1", t]->Value2 = "Daromatlar"; List-
>Range["D2", t]->Value2 = daromat[0];
52. List->Range["E1", t]->Value2 = "Soft Daromatlar"; List-
>Range["E2", t]->Value2 = Sof_daromat[0];
53. //Grafika qurish uchun qiymatlar diapazonini berish:
54. Grafik->SetSourceData(List->Range["A2", "E2"],
XLRowCol::xlColumns);
55. }
56. if(n>=2){List->Range["A3", t]->Value2 = "Febral"; List-
>Range["B3", t]->Value2 = maxsulot_soni[1];
57. List->Range["C3", t]->Value2 = harajat[1]; List-
>Range["D3", t]->Value2 = daromat[1];
58. List->Range["E3", t]->Value2 = Sof_daromat[1];

```

```

59. //Grafika qurish uchun qiymatlar diapazonini berish:
60. Grafik->SetSourceData(List->Range["A2", "E3"],
    XlRowCol::xlColumns);
61. }
62. if(n>=3){
63. List->Range["A4", t]->Value2 = "Mart"; List-
    >Range["B4", t]->Value2 = maxsulot_soni[2];
64. List->Range["C4", t]->Value2 = harajat[2]; List-
    >Range["D4", t]->Value2 = daromat[2];
65. List->Range["E4", t]->Value2 = Sof_daromat[2];
66. //Grafika qurish uchun qiymatlar diapazonini berish:
67. Grafik->SetSourceData(List->Range["A2", "E4"],
    XlRowCol::xlColumns);
68. }
69. if(n>=4){
70. List->Range["A5", t]->Value2 = "Aprel"; List-
    >Range["B5", t]->Value2 = maxsulot_soni[3];
71. List->Range["C5", t]->Value2 = harajat[3]; List-
    >Range["D5", t]->Value2 = daromat[3];
72. List->Range["E5", t]->Value2 = Sof_daromat[3];
73. //Grafika qurish uchun qiymatlar diapazonini berish:
74. Grafik->SetSourceData(List->Range["A2", "E5"],
    XlRowCol::xlColumns);}
75. if(n>=5){
76. List->Range["A6", t]->Value2 = "May"; List->Range["B6",
    t]->Value2 = maxsulot_soni[4];
77. List->Range["C6", t]->Value2 = harajat[4]; List-
    >Range["D6", t]->Value2 = daromat[4];
78. List->Range["E6", t]->Value2 = Sof_daromat[4];
79. //Grafika qurish uchun qiymatlar diapazonini berish:
80. Grafik->SetSourceData(List->Range["A2", "E6"],
    XlRowCol::xlColumns);}
81. if(n>=6){
82. List->Range["A7", t]->Value2 = "Iyun"; List->Range["B7",
    t]->Value2 = maxsulot_soni[5];
83. List->Range["C7", t]->Value2 = harajat[5]; List-
    >Range["D7", t]->Value2 = daromat[5];
83.2. List->Range["E7", t]->Value2 = Sof_daromat[5];
84. //Grafika qurish uchun qiymatlar diapazonini berish:
85. Grafik->SetSourceData(List->Range["A2", "E7"],
    XlRowCol::xlColumns);}
86. if(n>=7){

```

```

87. List->Range["A8", t]->Value2 = "Iyul"; List->Range["B8",
    t]->Value2 = maxsulot_soni[6];
88. List->Range["C8", t]->Value2 = harajat[6]; List-
    >Range["D8", t]->Value2 = daromat[6];
89. List->Range["E8", t]->Value2 = Sof_daromat[6];
90. //Grafika qurish uchun qiymatlar diapazonini berish:
91. Grafik->SetSourceData(List->Range["A2", "E8"],
    XlRowCol::xlColumns);}
92. if(n>=8){
93. List->Range["A9", t]->Value2 = "Avgust"; List-
    >Range["B9", t]->Value2 = maxsulot_soni[7];
94. List->Range["C9", t]->Value2 = harajat[7]; List-
    >Range["D9", t]->Value2 = daromat[7];
95. List->Range["E9", t]->Value2 = Sof_daromat[7];
96. //Grafika qurish uchun qiymatlar diapazonini berish:
97. Grafik->SetSourceData(List->Range["A2", "E9"],
    XlRowCol::xlColumns);}
98. if(n>=9){
99. List->Range["A10", t]->Value2 = "Sentyabr"; List-
    >Range["B10", t]->Value2 = maxsulot_soni[8];
100. List->Range["C10", t]->Value2 = harajat[8]; List-
    >Range["D10", t]->Value2 = daromat[8];
101. List->Range["E10", t]->Value2 = Sof_daromat[8];
102. //Grafika qurish uchun qiymatlar diapazonini berish:
103. Grafik->SetSourceData(List->Range["A2", "E10"],
    XlRowCol::xlColumns);}
104. if(n>=10){
105. List->Range["A11", t]->Value2 = "Oktyabr"; List-
    >Range["B11", t]->Value2 = maxsulot_soni[9];
106. List->Range["C11", t]->Value2 = harajat[9]; List-
    >Range["D11", t]->Value2 = daromat[9];
107. List->Range["E11", t]->Value2 = Sof_daromat[9];
108. //Grafika qurish uchun qiymatlar diapazonini berish:
109. Grafik->SetSourceData(List->Range["A2", "E11"],
    XlRowCol::xlColumns);}
110. if(n>=11){
111. List->Range["A12", t]->Value2 = "Noyabr"; List-
    >Range["B12", t]->Value2 = maxsulot_soni[10];
112. List->Range["C12", t]->Value2 = harajat[10]; List-
    >Range["D12", t]->Value2 = daromat[10];
113. List->Range["E12", t]->Value2 = Sof_daromat[10];
114. //Grafika qurish uchun qiymatlar diapazonini berish:

```

```

115. Grafik->SetSourceData(List->Range["A2", "E12"],
    XlRowCol::xlColumns);}
116. if(n==12){
117. List->Range["A13", t]->Value2 = "Dekabr"; List-
    >Range["B13", t]->Value2 = maxsulot_soni[11];
118. List->Range["C13", t]->Value2 = harajat[11]; List-
    >Range["D13", t]->Value2 = daromat[11];
118.2. List->Range["E13", t]->Value2 = Sof_daromat[11];
118.3. //Grafika qurish uchun qiymatlar diapazonini
    berish:
119. Grafik->SetSourceData(List->Range["A2", "E13"],
    XlRowCol::xlColumns);
120. }
121. //Grafikni turini berish("stolbikovaya diagramma"
    (gistogramma)):
122. Grafik->ChartType = XlChartType::xlColumnClustered;
123. //Grafikni legendasini ekranga chiqarish:
124. Grafik->HasLegend = true;
125. // Grafikning sarlavhasini ekranga chiqarish:
126. Grafik->HasTitle = true;
127. Grafik->ChartTitle->Caption = "Sotib olingan maxsulotlar
    harajatlari daromat va soft daromati "+n.ToString()+
    oylik korsatgichlari";
128. // osi X bo'yicha imzo:
129. Axis ^ GorizontalnayaOs = (Axis^)Grafik->Axes(
130. XlAxisType::xlCategory, XlAxisGroup::xlPrimary);
131. GorizontalnayaOs->HasTitle = true;
132. GorizontalnayaOs->HasMajorGridlines=true;
133. GorizontalnayaOs->AxisTitle->Text = "Oy nomlari";
134. // osi Y bo'yicha imzo:
135. Axis ^ VertikalnayaOs = (Axis^)Grafik->Axes(
136. XlAxisType::xlValue, XlAxisGroup::xlPrimary);
137. VertikalnayaOs->HasTitle = true;
138. VertikalnayaOs->AxisTitle->Text = "Ko`rsatgichlar";
139. //Grafikni quyidagi kengaytmali fayl ko'rinishida
    saqlash:
140. XL1->ActiveChart->Export("C:\\ExcelGrafik.jpg",t,t);
141. } }
141.2. } else {Console::Write("1 yilda 12 oy bor
    boshqattan urunib ko`ring"); goto bir;}
142. return 0;
143. }

```


Dastur tahlili:

Dasturda C++ ning kiritish (**cin**) operatori, **Visual C++** ning chiqarish (**Console::WriteLine()**) operatori, **Excel::Application** sinfidan nusxa olish (**_Application ^ XL1 = gnew Application();**) operatori, **MS Excell** da yangi varoq yaratish(**Workbook**) operatori, Varoqni faylda e'lon qilish (**Sheets**) operatori, **Excell** Listini tanlash (**_Worksheet**) operatori, Diagrammani yaratish(**_Chart**) operatori va diagrammani turli parametrlarini o'zgarturuvchi bir nechta funksiyalar ishlatilgan. Algoritmi quyidagicha: 4 ta massiv (**harajat[12]**, **daromat[12]**, **Sof_daromat[12]**, **maxsulot_soni[12]**) yaratilgan, ushbu massivlarga konsol muhitidan qiymat o'zlashtiriladi, ushbu massiv qiymatlarini **Excell** ning Listidagi katakchalarga tartib bilan yoziladi, kataklar diapazoni belgilanadi va (**_Chart**) operatori yordamida grafik quriladi.

7.7. Visual C++ ning Windows Application muhitida komponentalarning joylashish vaziyatlarini nazorat qilish

Ishning maksadi

Visual C++ muharririda komponentalarni joylashish pozitsiyalarini o'rnatuvchi vizual dasturlar yaratish.

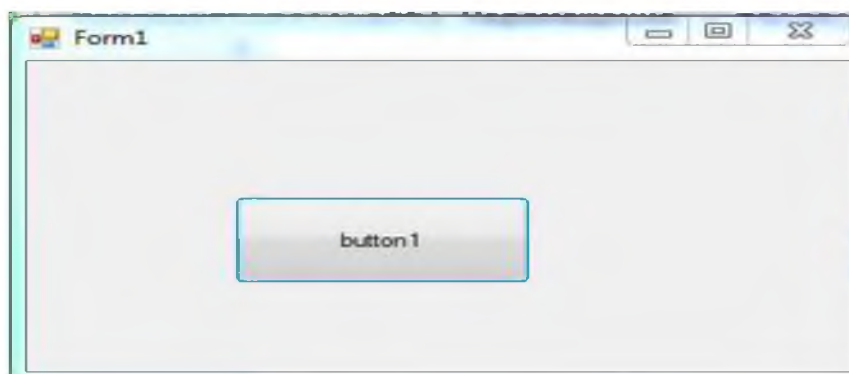
1- dastur: Formada komponentalar dasturning tashqi ko'rinishlarini, unumdorligini oshirishda muhim ahamiyat kasb etadi. Bizga ma'lumki dasturda ishlatiladigan komponentalarning joylashish pozitsiyalarini kompilyatsiyadan oldin o'zgartirish mumkin. **Visual C++** muharririda kompilyadan keyin ham o'zgartirish imkoniyati mavjud. Hozirgi yaratadigan dastur **Button** komponentasini forma bo'ylab ko'chirish imkoniyatlarini ochib beradi. Buning uchun **Windows Form Application** muhitida yangi loyiha yaratiladi. Formaga 1 ta **Button** komponentasi joylashtiriladi. Dastur kodlari quyidagicha:

```

1. #pragma endregion
2. bool Peremeçeniye;
3. int MouseDownX;
4. int MouseDownY;
5. private: System::Void Form1_Load(System::Object^ sender,
   System::EventArgs^ e) {
6. Peremeçeniye = false;}
7. private: System::Void button1_MouseDown(System::Object^
   sender, System::Windows::Forms::MouseEventArgs^ e) { if
   (e->Button == System::Windows::Forms::
8. MouseButton::Left) {
9. Peremeçeniye = true;
10. MouseDownX = e->X;
11. MouseDownY = e->Y;} }
12. private: System::Void button1_MouseUp(System::Object^
   sender, System::Windows::Forms::MouseEventArgs^ e) {
13. if (e->Button == System::Windows::Forms::
14. MouseButton::Left) Peremeçeniye = false;}
15. private: System::Void button1_MouseMove(System::Object^
   sender, System::Windows::Forms::MouseEventArgs^ e) {
16. if (Peremeçeniye == true)
17. {
18. auto Tochka = System::Drawing::Point();
19. Tochka.X = this->button1->Location.X + (e->X -
   MouseDownX);
20. Tochka.Y = this->button1->Location.Y + (e->Y -
   MouseDownY);
21. this->button1->Location = Tochka;
22. }
23. }

```

Dasturning tashqi ko‘rinishi quyidagicha:



7.14- rasm. Dastur natijasi

Dastur tahlili:

Dasturda 3 ta global o'zgaruvchilar (**bool Peremezeniye;int MouseDownX;int MouseDownY;**) e'lon qilindi. **bool Peremezeniye** – mantiqiy o'zgaruvchi bo'lib, komponentani ko'chirishga ruxsat bor yoki yo'qligini aniqlash uchun ishlatildi. Agar foydalanuvchi **Button1** tugmasini sichqonchaning chap tugmasi bilan bossa (**MouseDown, e->Button == Left**), u holda **Peremezeniye = true** bo'ladi. Bu esa Button1 tugmasini ko'chirish mumkinligiga anglatadi. Agar foydalanuvchi sichqonchaning tugmasini qo'yib yuborsa (Vaziyat **MouseUp**) u holda **Peremezeniye = true** bo'ladi. Qolgan 2 ta (**int MouseDownX; int MouseDownY;**) o'zgaruvchilar joriy vaziyatni to'ldirish uchun mo'ljallangan.

2- dastur: Operatsion tizimning maxsus ovozli fayllarini yuklovchi vuzual dastur.

Bizga ma'lumki operatsion tizimda ishlaganda xatoliklar, amallarni yoki operatsiyalarni bajarilishi, dasturlarni o'rnatish va x.k. lar haqida ovozlarni eshitiladi. Ushbu ovozli fayllardan joriy dasturimizda foydalanishga xarakat qilinadi. Buning uchun **Windows Form Application** muhitida yangi loyiha yaratiladi. Formaga 6 ta **Button** komponentasi joylashtiriladi. Dastur kodlari quyidagicha bo'ladi:

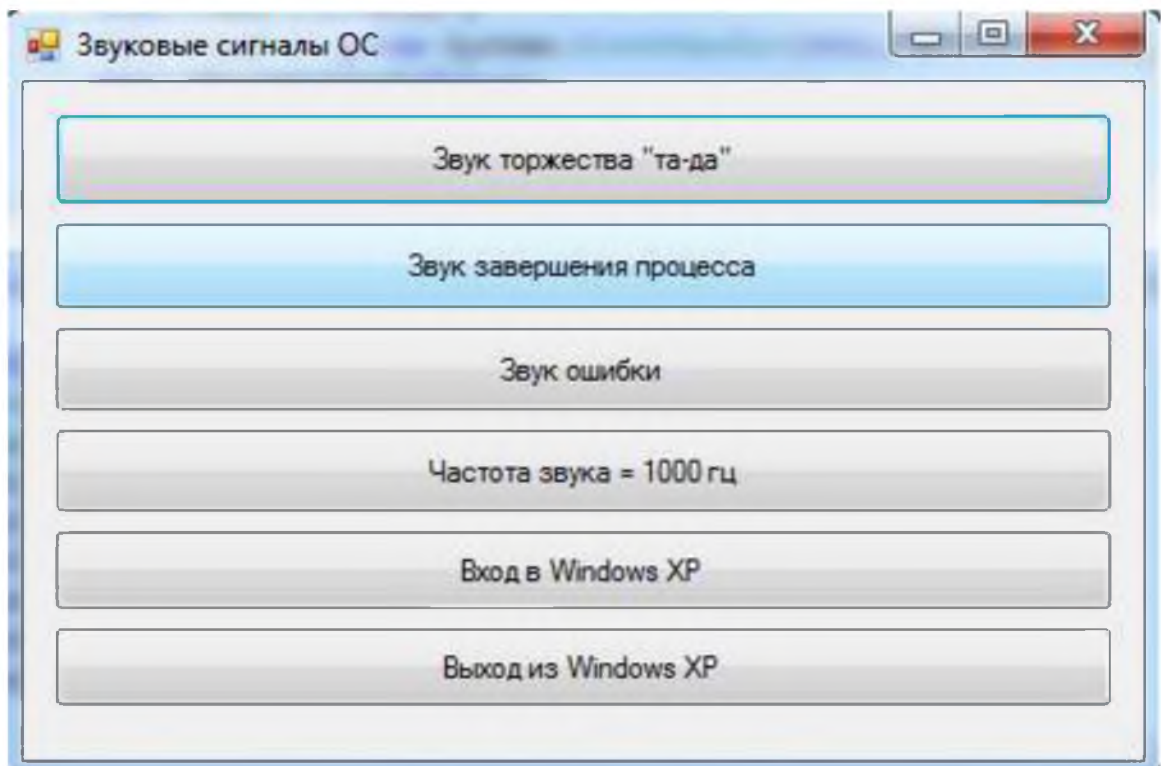
1. **#pragma endregion**
2. **System::Media::SoundPlayer ^ Pleyer;**
3. **private: System::Void Form1_Load(System::Object^ sender, System::EventArgs^ e) {**
4. **this->Text = "Звуковые сигналы OS";**
5. **Pleyer = gcnew System::Media::SoundPlayer();**
6. **// Zadayem nazvaniya komandnykh knopok:**
7. **button1->Text = "Звук торжества \"ta-da\"";**
8. **button2->Text = "Звук завершения протсесса";**
9. **button3->Text = "Звук oshibki";**
10. **button4->Text = "Chastota zvuka = 1000 gs";**
11. **button5->Text = "Vxod v Windows XP";**

```

12. button6->Text = "Выход из Windows XP";}
13. private: System::Void button1_Click(System::Object^
    sender, System::EventArgs^ e) {
14. //Vosproizvedeniye zvukovogo WAV-fayla
15. Pleyer->SoundLocation = "c:\\windows\\media\\tada.wav";
16. Pleyer->Play();
17. }
18. private: System::Void button2_Click(System::Object^
    sender, System::EventArgs^ e) {
19. // Klass SystemSounds poluchayet zvuki, svyazannye s
    naborom
20. // tipov zvukovykh sobytiy operatsionnoy sistemy Windows:
21. System::Media::SystemSounds::Asterisk->Play();
22. }
23. private: System::Void button3_Click(System::Object^
    sender, System::EventArgs^ e) {
24. System::Media::SystemSounds::Beep->Play();
25. }
26. private: System::Void button4_Click(System::Object^
    sender, System::EventArgs^ e) {
27. // Zvukovoy signal chastotoy 1000 Gts i dlitelnostyu
    0::5 sekund:
28. Console::Beep(1000, 500);}
29. private: System::Void button5_Click(System::Object^
    sender, System::EventArgs^ e) {
30. Pleyer->SoundLocation = "C:\\windows\\media\\Windows
    Startup.wav";
31. Pleyer->Play();}
32. private: System::Void button6_Click(System::Object^
    sender, System::EventArgs^ e) {
33. Pleyer->SoundLocation = "C:\\windows\\media\\Windows
    User Account //Control.wav";
34. Pleyer->Play();
35. }

```

Dasturning tashqi ko‘rinishi quyidagicha bo‘ladi:



5.15- rasm. Dastur natijasi

4- dastur: Operatsion tizimning video fayllarini yuklovchi dastur.

Ushbu dasturni tuzishda **Windows Media Player** dasturi ishlatiladi. Buni amalga oshirish uchun **Windows Form Application** muhitida yangi loyiha yaratiladi. Formaga **menuStrip** va **Windows Media Player** komponentalari joylashtiriladi. Lekin **ToolBox** asboblari panelida **Windows Media Player** komponentasi yo'q. Uni o'rnatish uchun **ToolBox** asboblari panelida sichqonchani o'ng tugmasi bosiladi va kontekst menyudan **Choose Items** buyrug'i tanlanadi. Ochilgan oynaning **SOM** bo'limidan **Windows.Media.Player** ni tanlab, **OK** tugmachasi bosiladi. Natijada **ToolBox** asboblari panelida ushbu komponenta paydo bo'ladi. **menuStrip** komponentasini sozlash 10.5-rasmda keltirilgan.

Dastur kodlari quyidagicha:

1. `#pragma endregion`
2. `OpenFileDialog ^ openFileDialog1;`

```

3. private: System::Void Form1_Load(System::Object^ sender,
   System::EventArgs^ e) {
4. openFileDialog1 = gcnew OpenFileDialog();
5. // PLYeYeRni versiyasi
6. this->Text = "Windows Media Player, versiya = " +
7. axWindowsMediaPlayer1->versionInfo;
8. }
9. private: System::Void
   ochishToolStripMenuItem_Click(System::Object^ sender,
   System::EventArgs^ e) { // PUNKT MYeNYuSI
10. // fOYDALANUVChI FAYLNI TANLAYDI :
11. openFileDialog1->ShowDialog();
12. // FAYLNING NOMINI POLYeYeRGA JO‘NATISH
13. axWindowsMediaPlayer1->URL = openFileDialog1->FileName;
14. // axWindowsMediaPlayer1->URL =
   "C:\\WINDOWS\\Media\\tada.wav";
15. // fayla FAYLNI YuKLASH KOMANDASI
16. axWindowsMediaPlayer1->Ctlcontrols->play();
17. // axWindowsMediaPlayer1->openPlayer(openFileDialog1-
   >FileName);
18. }
19. private: System::Void
   yopishToolStripMenuItem_Click(System::Object^ sender,
   System::EventArgs^ e) {
20. // KIRISH MYeNYu bo‘limi
21. Application::Exit();
22. }
23. private: System::Void
   ekranBoylabToolStripMenuItem_Click(System::Object^
   sender, System::EventArgs^ e) {
24. // Ekran bo‘ylab MYeNYu bo‘oimi.
25. if (axWindowsMediaPlayer1->playState ==
26. WMPLib::WMPPlayState::wmppsPlaying)
27. axWindowsMediaPlayer1->fullScreen = true;
28. }
29. private: System::Void
   tuxtatishToolStripMenuItem_Click(System::Object^ sender,
   System::EventArgs^ e) {
30. // To‘xtatish MYeNYu bo‘limi
31. axWindowsMediaPlayer1->Ctlcontrols->pause();}

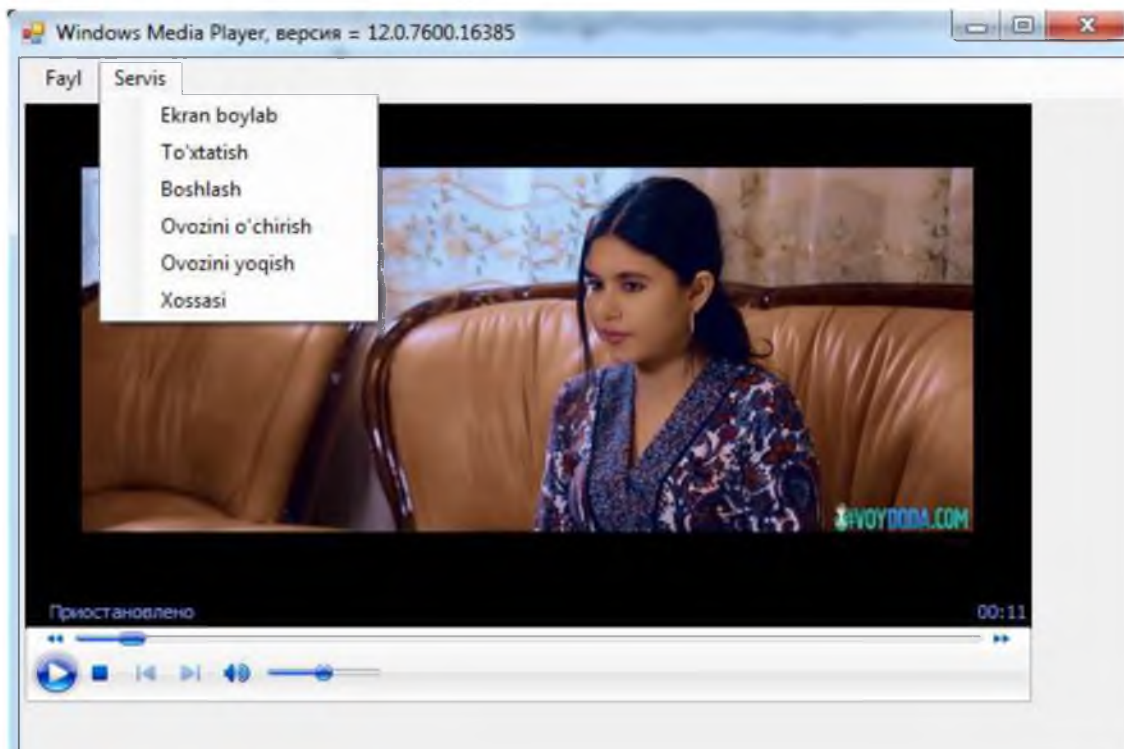
```

```

32. private: System::Void
    boshlashToolStripMenuItem_Click(System::Object^ sender,
    System::EventArgs^ e) {
33. // Boshlash MYeNYu bo'limi
34. axWindowsMediaPlayer1->Ctlcontrols->play();
35. }
36. private: System::Void
    ovoziniOchirishToolStripMenuItem_Click(System::Object^
    sender, System::EventArgs^ e) {
37. // Ovozni yoqish MYeNYu bo'limi
38. axWindowsMediaPlayer1->settings->mute = true;}
39. private: System::Void
    ovoziniYoqishToolStripMenuItem_Click(System::Object^
    sender, System::EventArgs^ e) {
40. // Ovozni o'chirish MYeNYu bo'limi
41. axWindowsMediaPlayer1->settings->mute = false;}
42. private: System::Void
    xossasiToolStripMenuItem_Click(System::Object^ sender,
    System::EventArgs^ e) {
43. // Fayl xossasi MYeNYu bo'limi
44. axWindowsMediaPlayer1->ShowPropertyPages();
45. }

```

Dasturning tashqi ko'rinishi quyidagicha bo'ladi:

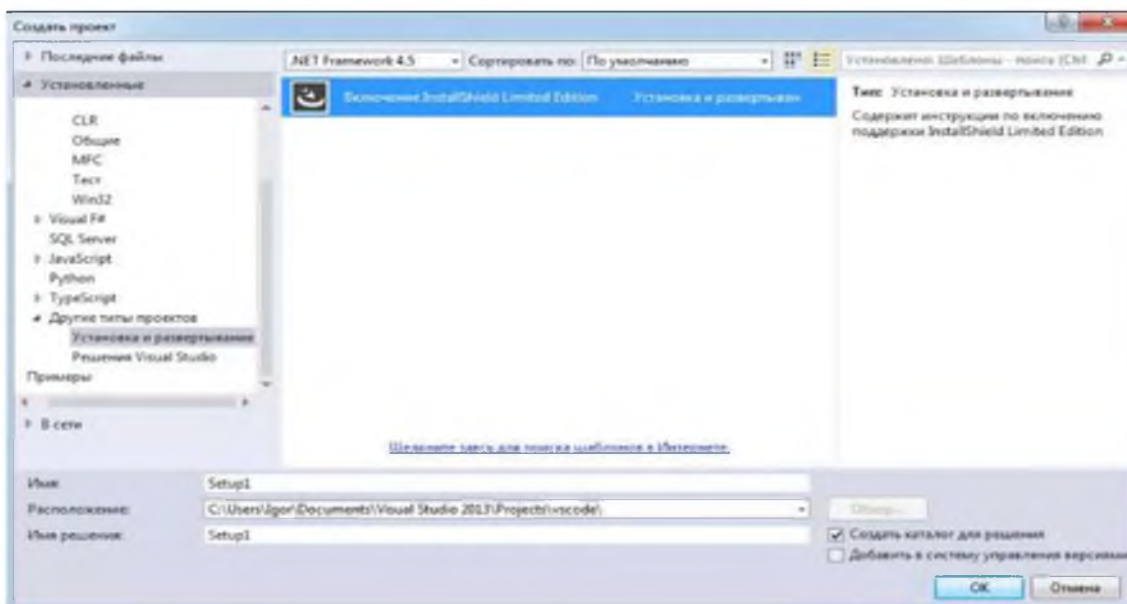


7.16- rasm. Windows Media Playerni yuklash

7.8. Loyihalarning yuklanuvchi “инсталляционные” dasturlar paketini yaratish

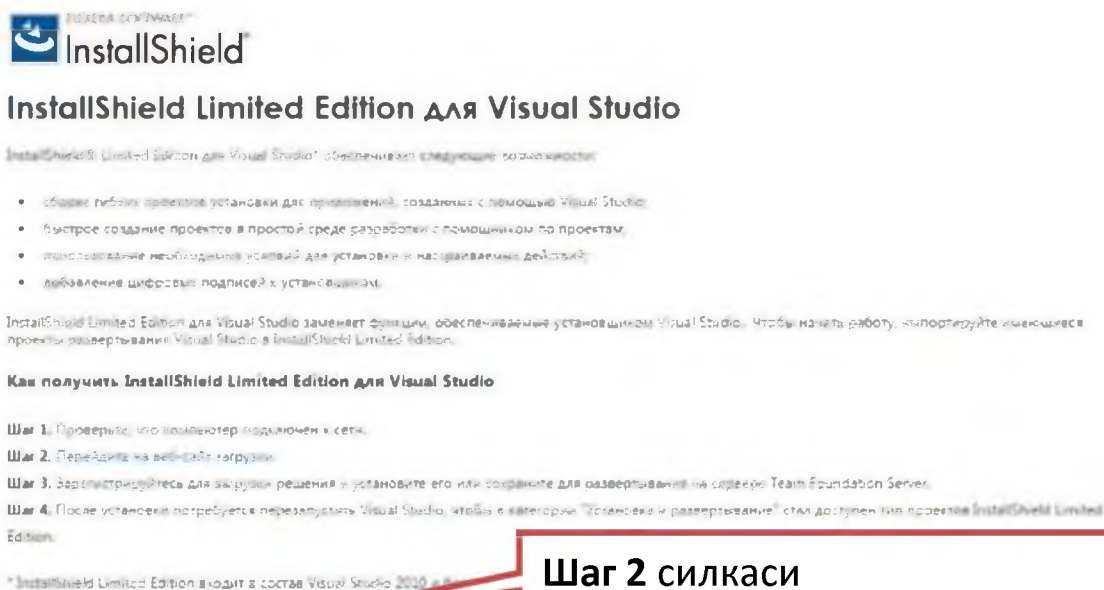
Bizga ma'lumki, avvalgi bo'limlarda yaratilgan kichik loyihalarni boshqa kompyuterlarda sinash yoki amaliyotga tadbiiq qilish uchun ularning yuklanuvchi paketlarini yaratish kerak. Yuklanuvchi paketlar - bu loyihaning dastur kodlari yaratuvchisida saqlangan holda uning “**exe**” formati (tizimga yuklanuvchi)dir. Yuklanuvchi paket o'zida bir nechta (**setup.exe**, ***.msi**, ***.dll**, ***.tlb**)kengaytmali fayllarni saqlaydi. Kichik loyihalarning operasion tizimga yuklanuvchi paketlarini yaratish uchun dasturlarni instalyasiyalovchi maxsus dasturlar bor. Masalan **InstallShield**, **InnoSetup** va boshqalar. Lekin bunday dasturlarni internet orqali yuklab olishni va aktivlashtirishni talab qiladi.

Yuklanuvchi paketlarni yaratish uchun **Visual Studio** dasturi ishga tushiriladi. Yangi loyiha yaratiladi(**New Project**), **Other Project Types** bo'limidan **Setup and Deployment (Установка и распространение)** qismiga o'tiladi va **Visual Studio Installer** va **Setup Project** tanlanadi. **Location** qismiga yuklanuvchi paket saqlanishi kerak bo'lgan diskdan joy ko'rsatiladi va OK tugmachasi bosiladi. Masalan: **S:\New_instal**. Misol uchun, bir loyihani yuklanuvchi paketini yaratib ko'raylik. “**Visual Studio**” dasturi ishga tushiriladi. Yangi loyiha yaratiladi “**New Project**”, “**Other Project Types**” bo'limidan “**Setup and Deployment**” (“**Установка и распространение**”) qismiga o'tiladi:



7.37-rasm. Yuklanuvchi paketlarni yaratishning 1- bosqichi

1-qadam. Veb brouzerda quyidagi sahifa ochiladi. Bu sahifadan “**Шаг 2**” havolasi bosiladi.



7.38-rasm. Yuklanuvchi paketlarni yaratishning 2- bosqichi

2-qadam.
Undan keyin anketa to‘ldiriladi.

* indicates required fields

First Name*

Last Name*

Email Address*

Important note: A serial number is required to activate the software. It will be emailed to this address.

Company*

Job Function*

Development Team Size*

Phone*

Country*

Postal Code*

Government or education organization

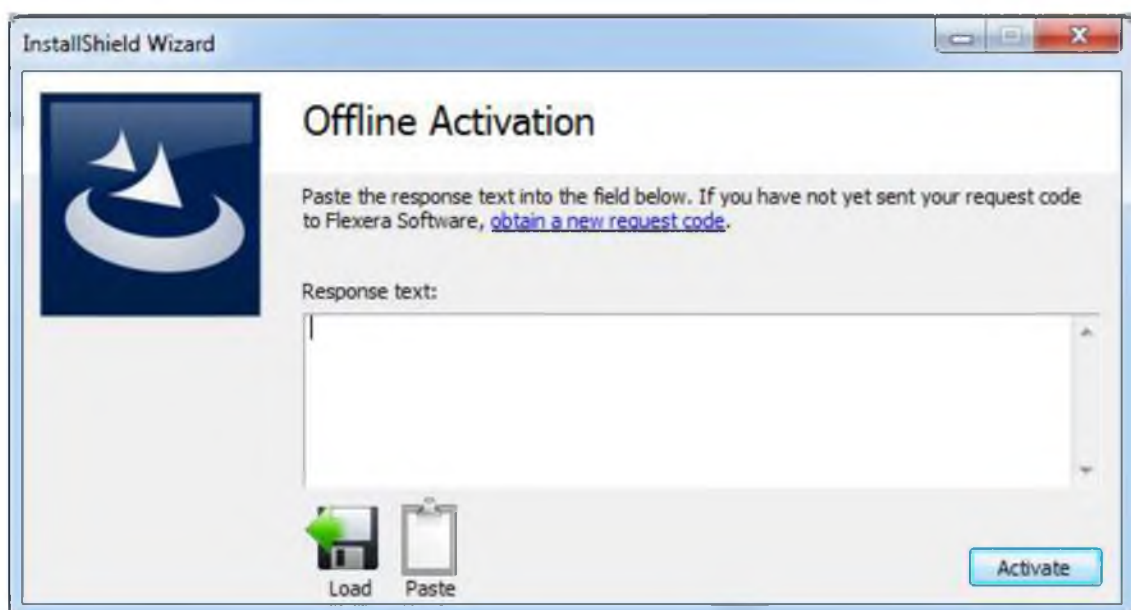
Register your email account for free community support?

Download Now >

7.39-rasm. Yuklanuvchi paketlarni yaratishning 3- bosqichi

3-qadam.

Anketa to'ldirilgandan keyin "**Download Now**" tugmasini bosib, dasturni yuklab olasiz va tizimga yuklaysiz. Yaratilgan tayyor loyiha dasturini ochasiz. Dastur ochilishida yangi yuklangan dastur lisenziya kalitini talab qiladi:



7.40-rasm. Yuklanuvchi paketlarni yaratishning 4- bosqichi

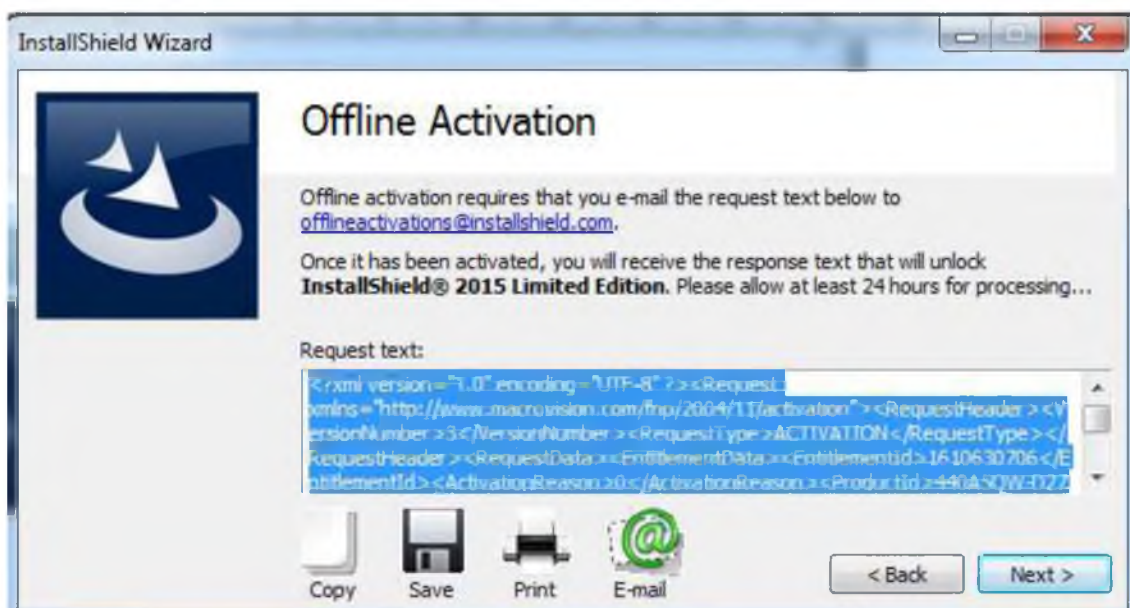
4-qadam.



7.41-rasm. Seriya raqamini kiritish oynasi

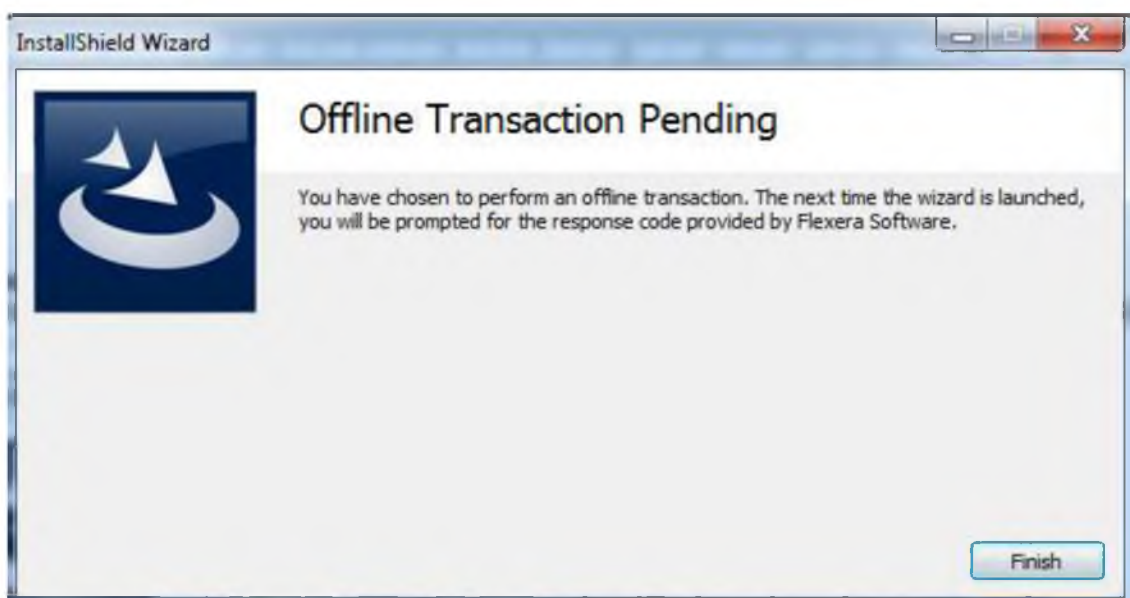
5-qadam.

Lisenziya kalitini anketangizda ko‘rsatilgan pochta manzilidan olinadi, so‘ralgan joyga yoziladi va “**Activate**” tugmachasi bosiladi.



7.42-rasm. Dasturni offline aktivlashtirish oynasi

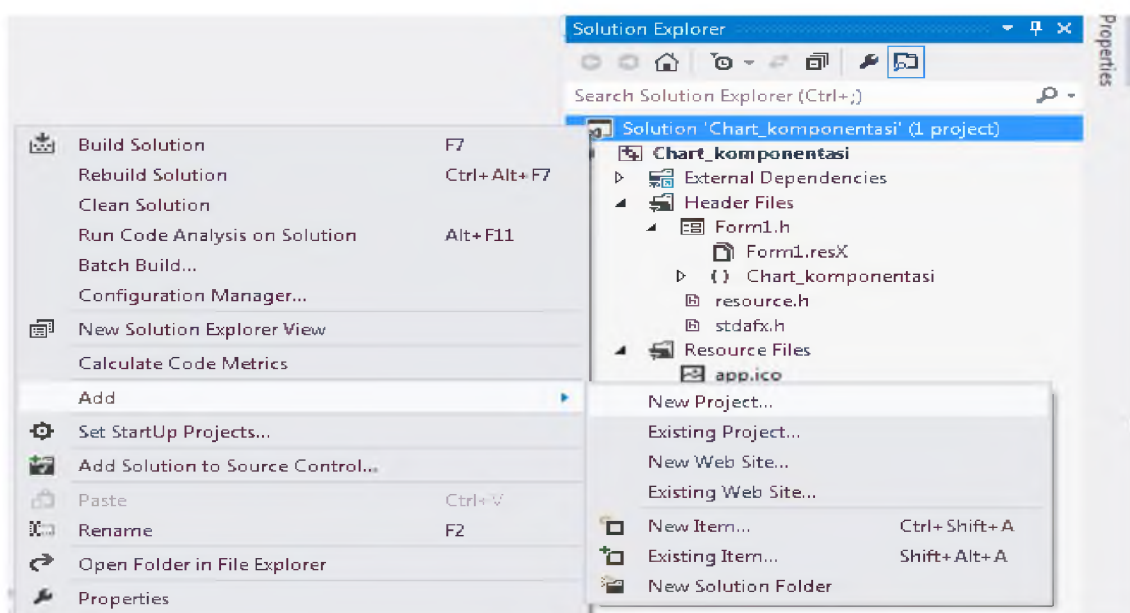
6- qadam.



7.43-rasm. Offline tranzaksiyani kutish oynasi

7-qadam.

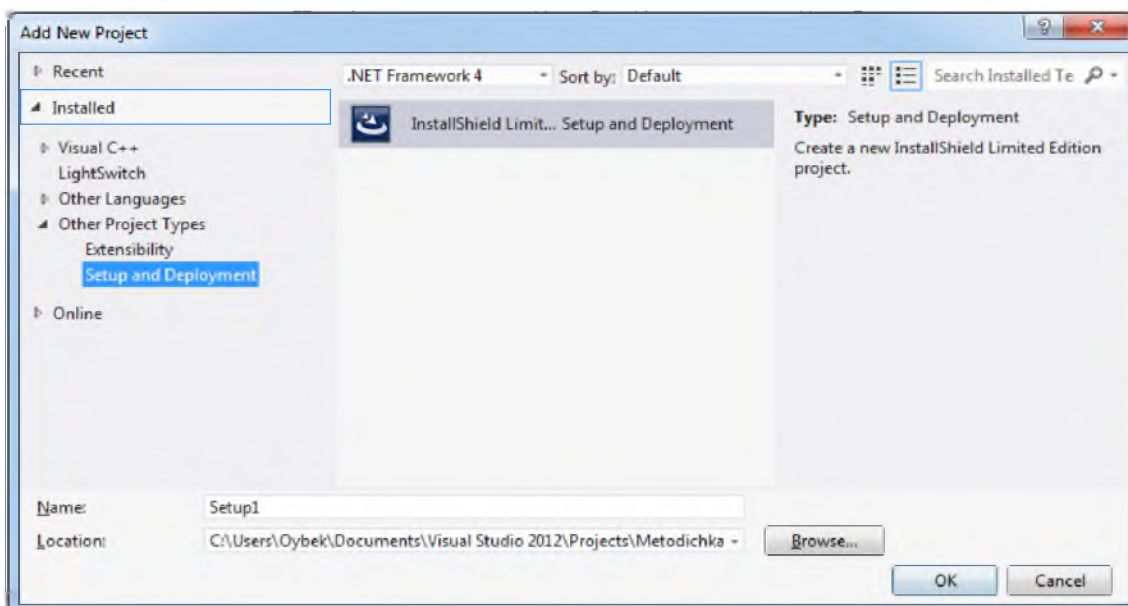
Finish tugmasi bosilgandan keyin. Yuklanuvchi paketini yaratish kerak bo'lgan dasturga yangi loyiha qo'shiladi: (**Solution Explorer**-> loyiha tanlanadi va sichqonchani o'ng tugmasini bosib, kontekst menyudan->**Add**->**New Project**..)



7.44-rasm. Yuklanuvchi dasturga qo'shiladigan yangi loyihani tanlash oynasi

8-qadam.

Yangi loyiha qo'shish oynasida **Other Project Types** bo'limidan **Setup and Deployment** qismiga o'tiladi va OK tugmasi bosiladi:



7.45-rasm. Yangi loyiha qo‘shish oynasi

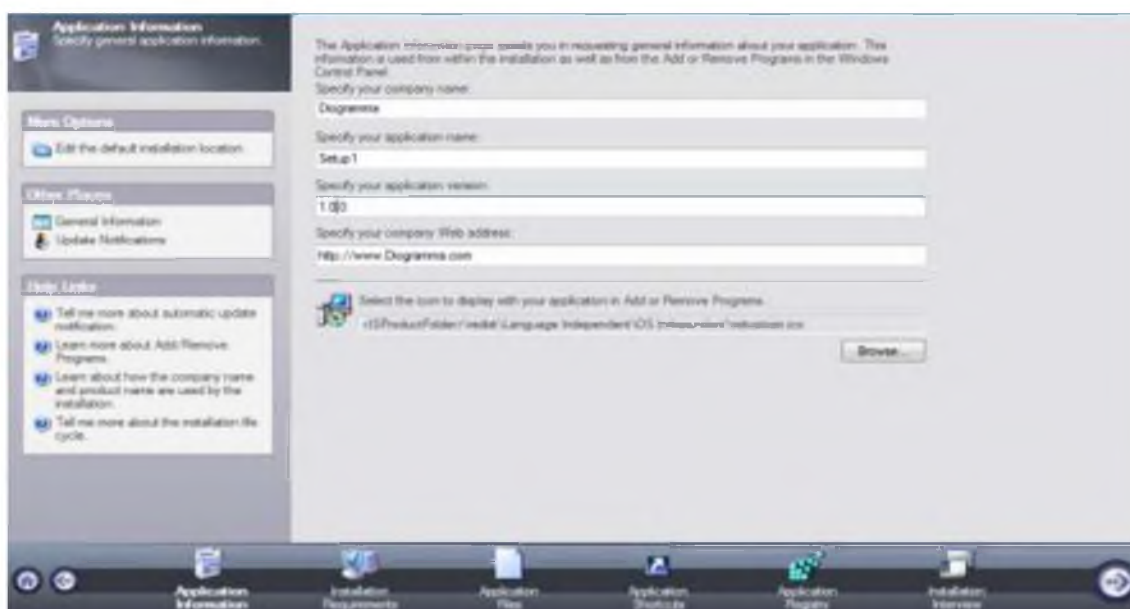
9-qadam. Ushbu jarayondan keyin quyidagi jarayon bajariladi:



7.46-rasm. Jarayon yordamchisi oynasi

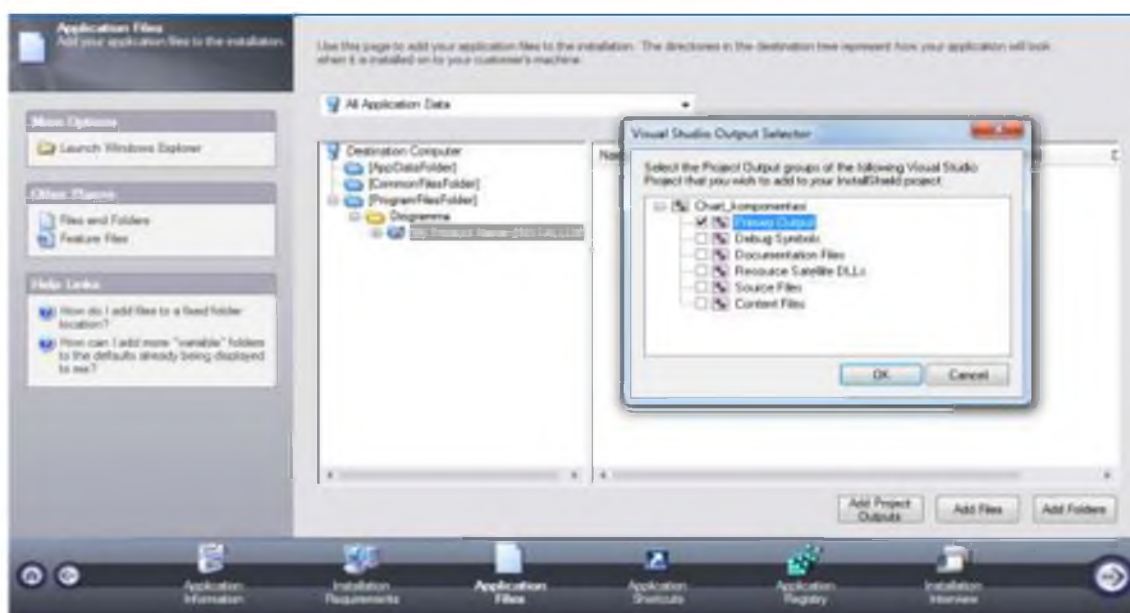
10-qadam.

“Далее” tugmasi bosiladi.



4.47-rasm. Loyiha to‘g‘risida ma‘lumot oynasi

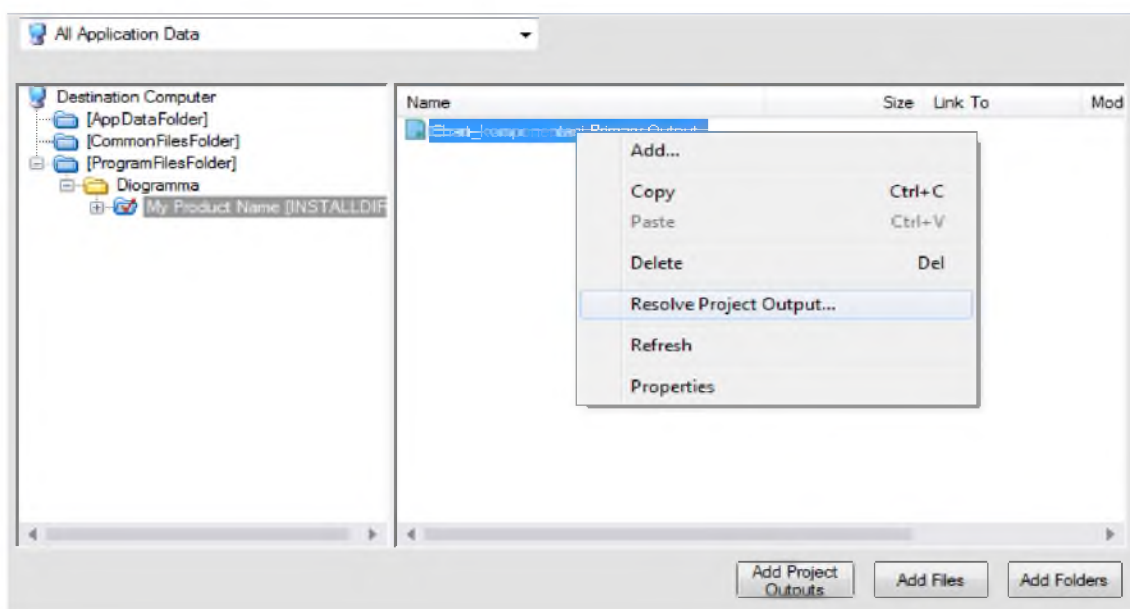
11-qadam. “Далее” tugmasi bosiladi.



7.48-rasm. Loyihaning fayllarini qo‘shish oynasi

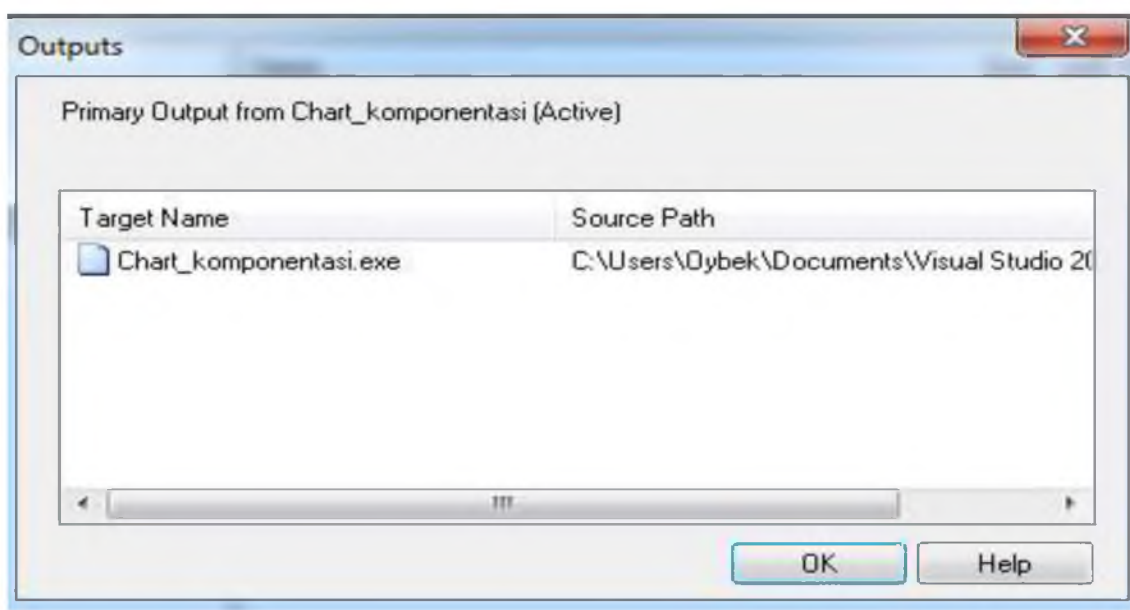
12-qadam.

“OK” va “Далее” tugmasi bosiladi.



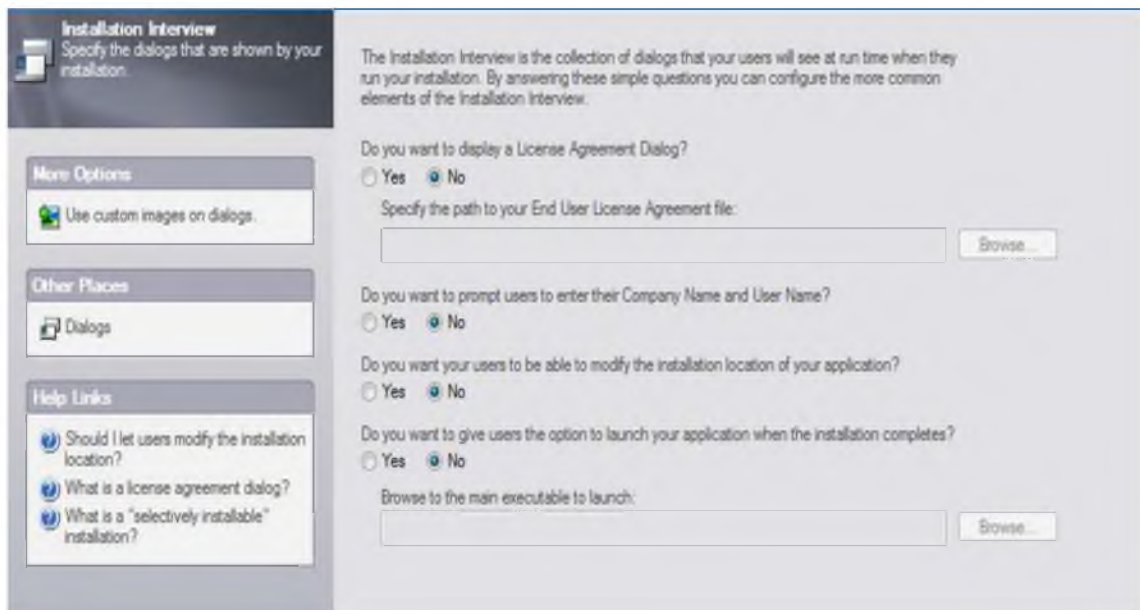
7.49-rasm. Yuklanuvchi paketlarni yaratishning 12- bosqichi

13-qadam. “OK” va “Далее” tugmasi bosiladi.



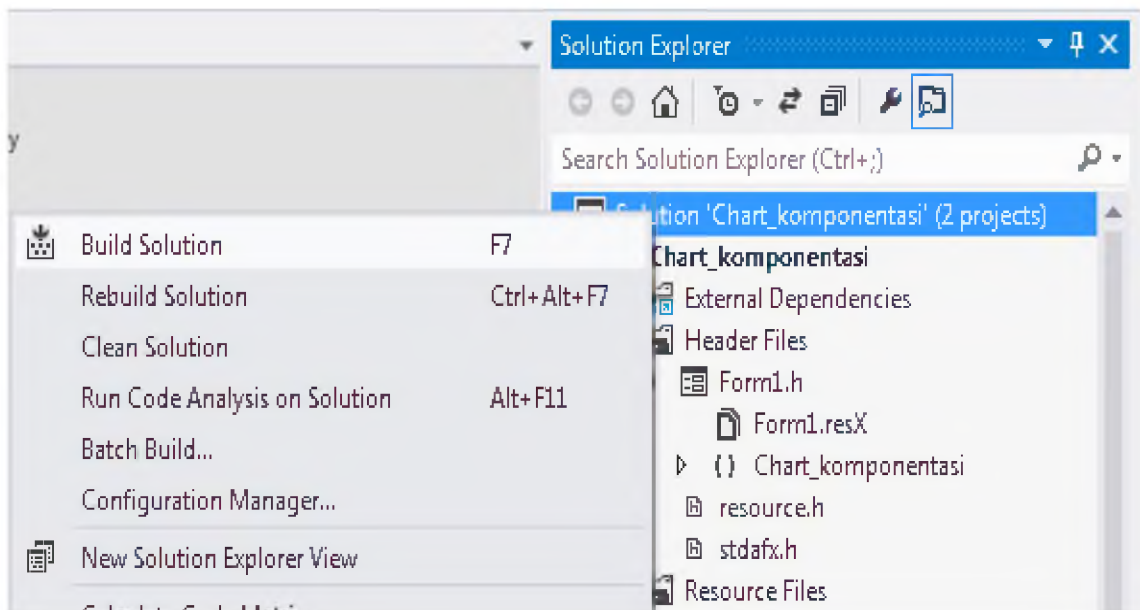
7.50-rasm. Yuklanuvchi paketlarni yaratishning 13- bosqichi

14-qadam. “OK” va “Далее” tugmasi bosiladi.



7.51-rasm. Yuklanuvchi paketlarni yaratishning 14- bosqichi

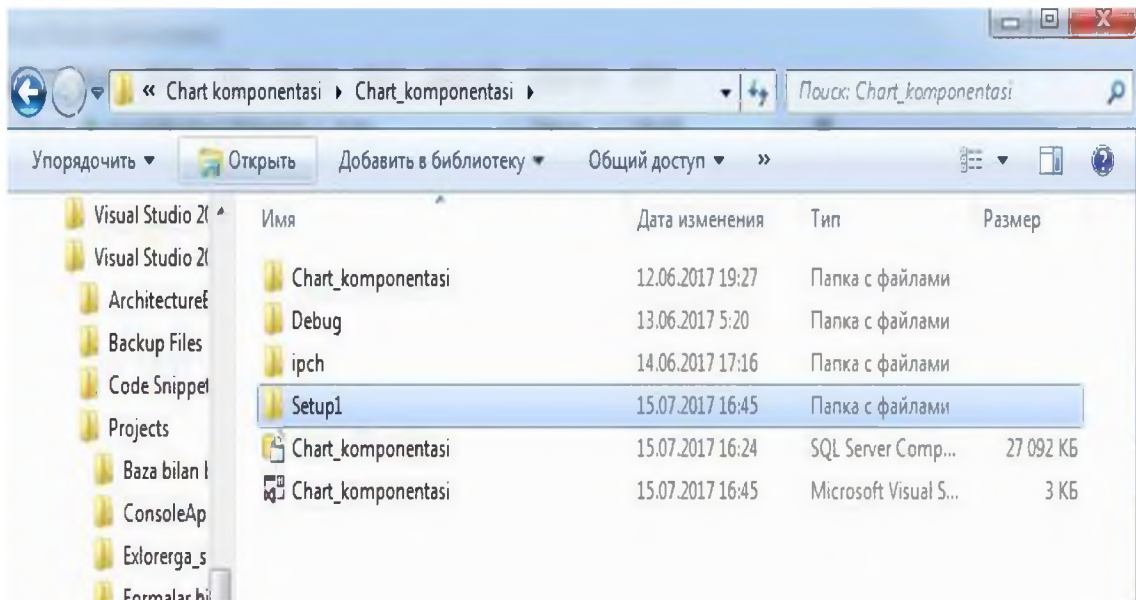
15-qadam. Kontekst menyudan **“Build Solution”** tanlanadi.



7.52-rasm. Yuklanuvchi paketlarni yaratishning 15- bosqichi

16-qadam.

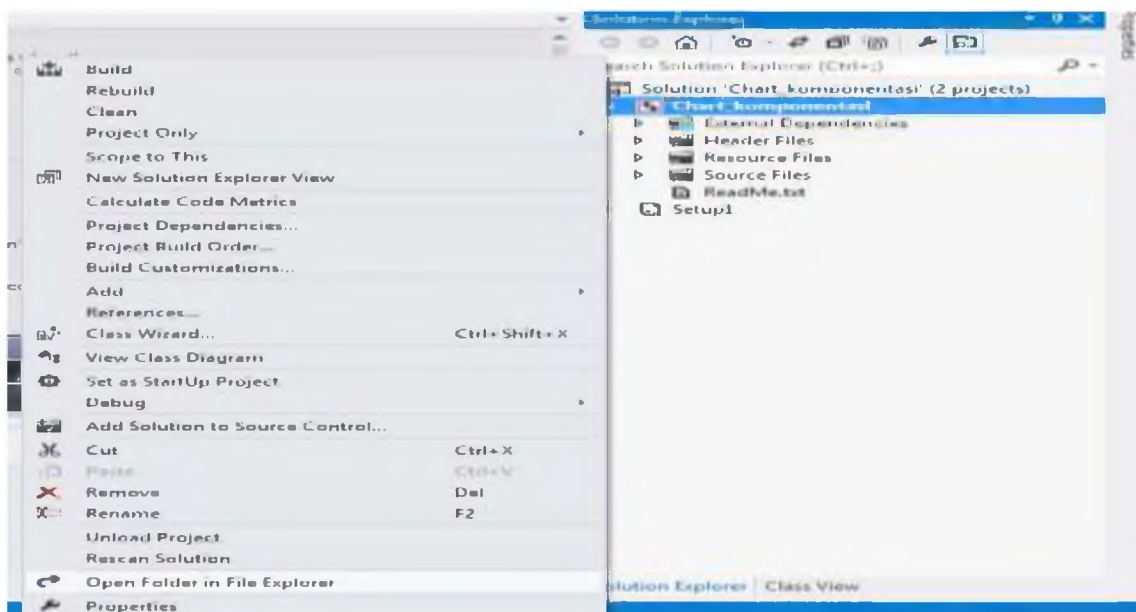
Bu jarayondan so‘ng yuklanuvchi paket saqlangan katalog ichida **setup1** nomli katalog paydo bo‘ladi.



7.53-рasm. Yuklanuvchi paket saqlangan joy

17-qadam.

Ushbu katalogga quyidagicha o'tsa ham bo'ladi:



7.54-рasm. Yuklanuvchi paket saqlangan joyga o'tish

18-qadam.

Setup1\Express\DVD-5\DiskImages\DISK1 ushbu manzilda yuklanuvchi fayl joylashgan. Natijada ushbu loyihaning yuklanuvchi paketini tarqatish imkoniyati vujudga keldi.

Takrorlash uchun savol va topshiriqlar:

1. **Visual C++** da **MS Excell** fayllarga eksport va import qanday amalga oshiriladi?
2. **Visual C++** da **MS Word** fayllarga eksport va import qanday amalga oshiriladi?
3. **Visual C++** da Diagrammalar bilan ishlash komponentalarini bilasizmi?
4. **Visual C++** da satrlar bilan ishlovchi qanday funksiyalar bor?
5. **Visual C++** da jadvalli tip nima?
6. **DataGridWiew** komponentasining vazifasi nima?
7. **Visual C++** da **ovozli** fayllarga qanday murojat qilinadi?
8. **Visual C++** da **video** fayllarga qanday murojat qilinadi?
9. **Visual C++** da rasmi fayllarga qanday murojat qilinadi?
10. **Visual C++** da dastur hisobodlari qanday tayyorlanadi?

ADABIYOTLAR

1. O‘zbekiston Respublikasini yanada rivojlantirish bo‘yicha harakatlar strategiyasi to‘g‘risida. O‘zbekiston Respublikasi Prezidentining PF-4947- son farmoni . Toshkent, 2017 yil 7 fevral.
2. Mirziyoev SH.M. Tanqidiy tahlil, qat‘iy tartib-intizom va shaxsiy javobgarlik - har bir rahbar faoliyatining kundalik qoidasi bo‘lishi kerak. Mamlakatimizni 2016 yilda ijtimoiy-iqtisodiy rivojlantirishning asosiy yakunlari va 2017 yilga mo‘ljallangan iqtisodiy dasturning eng muhim ustuvor yo‘nalishlariga bag‘ishlangan Vazirlar Mahkamasining kengaytirilgan majlisidagi ma‘ruza, 2017 yil 14 yanvar. - Toshkent: «O‘zbekiston», 2017.-104 b.
3. Mirziyoev SH.M. Buyuk kelajagimizni mard va olijanob xalqimiz bilan birga quramiz. 2017.
4. Mirziyoev SH.M. Qonun ustuvorligi va inson manfaatlarini ta‘minlash – yurt taraqqiyoti va xalq farovonligining garovi. 2017.
5. Xaydarova M.Yu., Mallayev O.U. Visual C++ da kichik loyihalar yaratish. O‘quv qo‘llanma. – T.: Aloqachi, 2019. – 224 b.
6. Nazirov SH.A., Qobulov R.V., Bobojanov M.R., Raxmanov Q.S. S va C++ tili. “Voriz- nashriyot” MCHJ, Toshkent 2013, 488 b.
7. Maxarov Q.T., Maxarov T.A. Visual Studio muhitida dasturlash asoslari(uslubiy qo‘llanma). Toshkent – 2017.
8. Maxarov Q.T., Nigmanova D.B., Navruzov E.R., Hayitqulov B.H. VC++ muhitida obyektlar bilan ishlash. Toshkent – 2017
9. Horstmann, Cay S. C++ for everyone/Cay S. Horstmann. Printed in the United States of America - 2nd ed. 2010. – P. 562.
10. Zubarov V.V. MS Visual C++ 2010 v srede .NET (2012) Piter.
11. Boris Paxomov. C/C++ i Visual C++ 2010 dlya nachinayuhix. - SBP: BXV- Peterburg. 2011.
12. J.Axmadaliev, R.Xoldorboev C++ dasturlash tilini o‘rganish bo‘yicha uslubiy qo‘llanma(2015).

13. Horton I.-Beginning Visual C++ 2012/ I.Horton. Published simultaneously in Canada.–2012. –P. 988.
14. Давыдов, В. Visual C++. Разработка Windows-приложений с помощью MFC и API-функций / В. Давыдов. - М.: БХВ-Петербург, 2014. - 576 с.
15. Довбуш, Галина Visual C++ на примерах / Галина Довбуш, Анатолий Хомоненко. - М.: БХВ-Петербург, 2012. - 528 с.
16. Зиборов, В. MS Visual C++ 2010 в среде .NET / В. Зиборов. - М.: Питер, 2012. - 320 с.
17. Мешков, А. Visual C++ и MFC / А. Мешков, Ю. Тихомиров. - М.: БХВ-Петербург, 2013. - 546 с.
18. Панюкова, Т. А. Языки и методы программирования. Создание простых GUI-приложений с помощью Visual C++. Учебное пособие / Т.А. Панюкова, А.В. Панюков. - Москва: Мир, 2015. - 144 с.
19. Пахомов, Б. C/C++ и MS Visual C++ 2010 для начинающих / Б. Пахомов. - М.: БХВ-Петербург, 2011. - 736 с.
20. Пахомов, Борис C/C++ и MS Visual C++ 2012 для начинающих / Борис Пахомов. - Москва: СИНТЕГ, 2015. - 518 с.
21. Пахомов, Борис C/C++ и MS Visual C++ 2012 для начинающих / Борис Пахомов. - М.: "БХВ-Петербург", 2013. - 502 с.
22. Понамарев, В. Программирование на C++/C# в Visual Studio .NET 2003 / В. Понамарев. - М.: БХВ-Петербург, 2015. - 917 с.
23. Сидорина, Татьяна Самоучитель Microsoft Visual Studio C++ и MFC / Татьяна Сидорина. - М.: "БХВ-Петербург", 2014. - 848 с.
24. <http://labs.org.ru/visual-c/>
25. <https://msdn.microsoft.com/ru-ru/library/60k1461a.aspx>
26. <https://professorweb.ru/my/programs/visual-studio/level1/>

MUNDARIJA

KIRISH.....	3
1. 1- LABORATORIYA ISHI. NOMALUM NOMLAR FAZOSI VA TIPLARNI DINAMIK TARZDA ANIQLASH	4
1.1. Nomaʼlum turlar va nomaʼlum nomlar fazosi	4
2. 2- LABORATORIYA ISHI. STL KUTUBXONALARI. KONTEYNER SINFLAR.	10
2.1. Chiziqli konteynerlar (array, vector, deque, list, forward_list)	
2.1.1. array sinfidan foydalanib, amaliy dasturlar yaratish	10
2.1.2. vector sinfidan foydalanib, amaliy dasturlar yaratish	14
2.1.3. deque konteyneridan foydalanib, amaliy dasturlar yaratish.	17
2.1.4. forward_list konteyneridan foydalanib, amaliy dasturlar yaratish	21
2.1.5. list konteyneridan foydalanib, amaliy dasturlar yaratish	25
2.2. Assotsiativ konteynerlar (set, map, multiset, multimap)	27
2.2.1. set va multiset sinflaridan foydalanib, amaliy dasturlar yaratish	27
2.2.2. map va multimap sinflaridan foydalanib, amaliy dasturlar yaratish.....	31
2.3. stack, queue, priority_queue konteyner adapterlari	
2.3.1. stack sinflaridan foydalanib, amaliy dasturlar yaratish.....	36
2.3.2. queue ba priority_queue sinflaridan foydalanib, amaliy dasturlar yaratish.....	40
2.4. Iteratorlar va ularning qoʻllanilishi. Xotirani taqsimlovchilar va ularga qoʻyilgan talablar.	43
2.4.1. Standart algoritmlar va iteratorlardan foydalanib, amaliy dasturlar yaratish.....	43
Takrorlash uchun savol va topshiriqlar.....	54
3. 3- LABORATORIYA ISHI. SONLI SINFLAR BILAN ISHLASH	55
3.1. Complex sonlar va ular bilan ishlashga doir amaliy dasturlar yaratish	55
4. 4- LABORATORIYA ISHI. SINTAKSIS TAHLIL	66
4. Ifodalarning sintaksis tahlili.....	66
4.1. Leksik tahlilni amalga oshiruvchi amaliy dastur tuzish	66
5. 5- LABORATORIYA ISHI.	

MURAKKAB SARALASH ALGORITMLARI VA KATTA SONLAR BILAN ISHLASH	95
5.1. Katta sonlar bilan ishlovchi kutubxona va uning funksiyalariga doir amaliy dasturlar tuzish.....	95
5.2. Murakkab saralash algoritmlari.....	102
6. 6- LABORATORIYA ISHI.	
VISUAL C++ MUHITIDA DASTURLASH	114
6.1. Visual C++ muhitida menyular va uskunalar paneli	114
6.2. Visual komponentalar yordamida amaliy dasturlar tuzish ..	131
6.3. Visual C++ muhitining komponentalari: CheskBox, ChekedListBox, ComboBox komponentalarining xossalari (Properties) va hodisalari(Events)	139
6.4. Visual C++ muhitida muloqot oynalari bilan ishlash: DateTimePiker, LinkLabel va ListBox komponentalarining xossalari (Properties) va hodisalari(Events).....	150
7. 7- LABORATORIYA ISHI.	
KICHIK LOYIHALAR YARATISH	166
7.1. Matnli maydonga parol kiritish, satrlarga qayta ishlov berish	166
7.2. Oddiy matn muxarriri. Fayl ochish va saqlash. Formaning Closing hodisasi.....	178
7.3. Jadvalli ma'lumotlar asosida va Chart komponentasi yordamida grafik diagrammalar yaratish.....	191
7.4. Veb brauzerda HTML jadvallarni tasvirlash va shakllantirish.....	201
7.5. Visual C++da MS word imkoniyatlaridan foydalanib, jadvallar yaratish, ularni word faylga eksport qilish va taqdim etish.....	207
7.6. Visual C++ da MS Excell imkoniyatlaridan foydalanib, diagrammalar yaratish va ularni turli kengaytmalarda saqlash.....	214
7.7. Visual C++ ning Windows Application muhitida komponentalarning joylashish vaziyatlarini nazorat qilish..	222
7.8. Loyihalarning yuklanuvchi “инсталляционные” dasturlar paketini yaratish	229
Takrorlash uchun savol va topshiriqlar	239
FOYDALANILGAN ADABIYOTLAR	240

**«Dasturlash 2» fanidan laboratoriya ishlarini bajarish
uchun uslubiy qo‘llanma**

Informatika asoslari kafedrasining
11.02.2020 yildagi 7A – sonli majlisining
bayonnomasiga asosan fakultet ilmiy - uslubiy
kengashiga muhokama uchun tavsiya etildi.

Kompyuter injiniringi fakulteti ilmiy - uslubiy
kengashining 26.05.2020 yildagi 21- sonli
bayonnomasiga asosan universitet ilmiy - uslubiy
kengashiga muhokama uchun tavsiya etildi.

TATU ilmiy-uslubiy kengashi majlisida
ko‘rib chiqildi va nashr etishra ruxsat etildi,
2020 yil 23 iyun 9(134) - sonli bayonnoma

Tuzuvchilar: O.U. Mallayev,
Z.Sh. Abdullayev,
Yu.Sh. Baxramova

Taqrizchi: B.B. Mo‘minov

Mas’ul muharrir: B.B. Mo‘minov

Musahhih: K.A.Gayubova

Mallayev O.U., Abdullayeva Z,Sh., Baxramova Yu.Sh.

«Dasturlash 2»

fanidan laboratoriya ishlarini bajarish uchun uslubiy qo‘llanma

**Бичими 60x84^{1/16}. Рақамли босма усули. Times гарнитураси.
Шартли босма табоғи: 15,25. Адади 50 нусха. Буюртма № 185.**

Гувоҳнома № 10-3719

**“Тошкент кимё технология институти” босмаҳонасида чоп этилган.
Босмаҳона манзили: 100011, Тошкент ш., Навоий кўчаси, 32-уй.**