

R.A.Ro'ziyev, SH.J.Xudoyorov, O'.M.Mirsanov

# Delphi dasturlash tilida massivlar bilan ishlash (uslubiy qo'llanma)



**O'ZBEKISTON RESPUBLIKASI XALQ TA'LIMI VAZIRLIGI**  
**NAVOIY DAVLAT PEDAGOGIKA INSTITUTI**

# ***Delphi dasturlash tilida massivlar bilan ishlash***

***(Uslubiy qo'llanma)***



**Navoiy-2013**

Ushbu uslubiy qoʻllanma akademik litsey, kasb-hunar kollejlari, oliy oʻquv yurtlari talabalari hamda mustaqil foydalanuvchilar uchun moʻljallangan.

**Tuzuvchilar:**

**R.A.Roʻziyev** – Informatika va axborot texnologiyalari kafedrası dotsenti.

**Sh.J.Xudoyorov** - f.-m.f.n., Informatika va axborot texnologiyalari kafedrası mudiri.

**Oʻ.M.Mirsanov** – Informatika va axborot texnologiyalari kafedrası oʻqituvchisi.

**Taqrizchilar:**

**M.B.Bozorov** –Navoiy DKI “Texnologik jarayonlar va ishlab chiqarishni avtomatlashtirish va boshqaruv” kafedrası mudiri, professor.

**G`R.Yodgorov** - f -m.f.n., Informatika va axborot texnologiyalari kafedrası dotsenti.

Uslubiy qoʻllanma Navoiy davlat pedagogika institutining 2013 yil \_\_\_ noyabrdagi \_\_\_ -sonli Ilmiy kengashida koʻrib chiqilgan va chop etishga tavsiya etilgan.

## MUNDARIJA

KIRISH.....	4
1-§. BIR O`LCHOVLI MASSIVLAR.....	5
2-§. IKKI O`LCHOVLI MASSIVLAR.....	35
FOYDALANILGAN ADABIYOTLAR.....	82

## KIRISH

Ko'p hollarda jadval yoki matritsalar ko'rinishidagi ma'lumotlar bilan ish yuritish kerak bo'ladi. Jadvalda ma'lumotlar juda ko'p bo'lgani sabab, ularning har bir yacheykasidagi sonni mos ravishda bitta o'zgaruvchiga qiymat qilib berilsa ular ustida ish bajarish ancha noqulayliklarga olib keladi. Shu sabab dasturlashda bunday muammolar massivlarni ishlatish yordamida hal qilinadi.

Uslubiy qo'llanma Delphi dasturlash tilida massivlar bilan ishlashga mo'ljallangan bo'lib, unda bir o'lchovli va ikki o'lchovli massivlarga doir bir qancha masalalar keltirilgan va ularni hal qilish uchun delphi muhitida amaliy dasturlar tuzib ko'rsatilgan, natijalar olinib ularning tahlili keltirilgan. Shu bilan birgalikda foydalanuvchilarga mustaqil yechish uchun misollar berilgan.

Ushbu uslubiy qo'llanma akademik litsey, kasb-hunar kollejlari, oliy o'quv yurti talabalari hamda mustaqil foydalanuvchilar uchun mo'ljallangan.

***Mualliflar***

## 1-§. BIR O`LCHOVLI MASSIVLAR

Massiv - bu bir nom bilan belgilangan qiymatlar guruhi yoki jadvaldir. Massivning har bir elementi massiv nomidan so'ng kvadrat qavs ichiga olingan raqam va arifmetik ifoda yozish bilan belgilanadi. Qavs ichidagi raqam massiv indeksini belgilaydi. Vektorni bir o'lchovli massiv, matritsani ikki o'lchovli massiv deb qarash mumkin.

Ma'lumki, Delphi dasturlash tilida ishlatiladigan ma'lumotlar ikki turga oid: oddiy(**Real, Integer, Boolean, Char**, qayd qilingan va chegaralangan) va murakkab turdagi ma'lumotlarga muntazam turdagi ma'lumotlar (yozuvlar, fayl turlari, to'plamlar va ko'rsatkichlar) ga bo'linadi. Oddiy turdagi ma'lumotlarning boshqa turdagi ma'lumotlardan asosiy farqlanuvchi belgisi ularning tartiblanganligi va yaxlitlanganligidir, ya'ni masalan, **INTEGER** turiga mansub ixtiyoriy kattalik alohida raqamlarga bo'linmaydigan yaxlit kattalikdir(**Integer** toifasidagi kattalik sifatida bitta butun son tushuniladi). Lekin sonli o'qning ixtiyoriy bo'lagini butun sonlar ketma-ketligida qarajak, u holda shu ketma-ketlik **i** raqami to'g'riisida gapirish, bu elementlar(butun sonlar) to'plamiga "butun sonlar" degan umumlashtiruvchi nom berish mumkin.

Delphi dasturlash tilida bunday kattaliklarni ma'lumotlarning muntazam turi ko'rinishida ifodalash mumkin. Umumiy nomga ega, tartiblangan kattaliklar ketma-ketligi *muntazam toifadagi ma'lumotlar yoki massivlar* deb yuritiladi. Ularning tashkil etuvchilari massiv elementlari va elementlari toifasi bazaviy toifa deb nomlanadi. Masalan, quyidagi tartiblangan haqiqiy sonlar ketma-ketligi 1.5,18.7, -5, 4.9, 0.88.

Haqiqiy sonlar massivi, bu massivning bazaviy turi esa haqiqiy(**Real**) toifadir.

Massiv tushunchasi A-umumiy nomga ega bo'lgan va bir toifadagi qo'zg'almas kattaliklar ketma-ketligidan iborat bo'lgan sonli vektor.

$$A(4)=(a1,a2,a3,a4)$$

misolida tushuntirish mumkin. Bu yerda  $a_1, a_2 \dots$  massiv elementlaridir. Ularni ifodalashda ko'rsatkichli (indeks) o'zgaruvchilardan foydalaniladi.

Matematika kursidan ma'lumki, ko'rsatkichli o'zgaruvchilarning tartiblangan ketma-ketlikdagi o'rnini bildirib, qavslar ichiga olinib yoki massiv nomidan birmuncha past ko'rsatilar edi. Masalan,  $A(1)$  yoki  $A_1$ , umumiy holda esa  $A_1$ , bu yerda  $i=1,2,3,\dots,n$ .

Delphi dasturlash tilidagi o'zgaruvchining ko'rsatkichi to'rtburchak qavslar ichiga olib yoziladi, ya'ni  $A[1]=1.6, A[2]=23.7, A[3]=-7, A[4]=9.5$ .

Agar dasturda massiv ishlatilayotgan bo'lsa, u holda uni o'zgaruvchilar bo'limi VAR yoki toifalar bo'limi TYPE da tasvirlash zarur, masalan, **Var** bo'limida massiv quyidagi ko'rinishda tasvirlanadi:

Var <massiv nomi>:

Array [<ko'rsatkich toifasi>] of <element toifasi>;

Bu yerda <massiv nomi> - ixtiyoriy idetifikator, Array(massiv) va of(dan)-xizmatchi so'zlar, <ko'rsatkich toifasi>-ko'rsatkichli ifoda, bu toifa qiymatlari massiv elementlarining sonni belgilaydi va ko'rsatkichlarni yozish uchun ishlatiladigan belgilar to'plamini ko'rsatadi, shuning uchun bu toifa sifatida **Real** va **Integer** toifasidan tashqari barcha oddiy toifalarni ishlatish mumkin, <element toifasi>-massiv elementlari toifasi bo'lib, bu toifa sifatida fayl va to'plam toifasidan tashqari barcha toifalarni ishlatish mumkin.

Yuqorida ko'rib o'tilgan A vektorini massiv ko'rinishida dasturda quyidagicha tasvirlash mumkin:

**Var a:array[1..5] of real;**

Indeks sifatida faqat aniq son emas, qiymati massiv elementlarining tartib raqamini belgilovchi ifoda ham ishlatilishi mumkin, masalan **A[i+5], B[I div(j+6)], C[n1 or n2], YEAR[1988]** va hokazo.

Ko'rsatkichli ifoda qiymatining toifasi massiv elementi ko'rsatkichining toifasi deyiladi va ularning tartiblanganligini aniqlovchi to'plam bo'lishi kerak. Quyida ko'rsatkich toifasi sifatida paskalda ishlatilishi mumkin bo'lgan toifalar bilan tanishamiz. Ma'lumki, paskal tilida **Integer** va **Real** toifasidagi ma'lumotlar to'plami

cheklanmagan, **Real** toifa esa, shuningdek, tartiblanmagandir ham. Shu sababli massiv ko'rsatkichi toifasi sifatida **Integer** va **Real** toifasiga mansub ma'lumotlardan foydalanish mumkin emas;

```
Var a:Array[1..4] of real;  
c:array[1..4] of integer;  
b:array[1..20,1..45] of integer;
```

Ushbu hollarda massiv ko'rsatkichi toifasi sifatida chegaralangan toifa, ko'pincha chegaralangan butun toifa ma'lumotlari ishlatiladi. Masalan, 100 ta haqiqiy elementlardan iborat massiv quyidagi ko'rinishda tavsiflanishi mumkin:

```
var a:array[1..10] of real;
```

Bu yerda 1 dan 10 gacha chegaralangan butun toifa **a** massiv elementlari soni 10 ta ekanligi va ularning tartiblanganligini bildiradi.

To'g'ri tavsiflangan massivlarga misollar:

```
var massiv:array[-745..-1] of real;  
das:array[1477..1988] of char;  
l:array[boolean]of char;  
a,b,c:array[1..50] of rael;
```

Massiv indeksi chegarasini butun tipdagi o'zgarmas bilan ham ifodalash mumkin, masalan, **const nmax=50;**

```
var a:array[1..nmax] of real;
```

Ma'lumki, qayd qilingan toifa qiymatlarining to'plami ham cheklangan va tartiblangan to'plamni tashkil qiladi, bu esa o'z navbatida, qayd qilingan toifa qiymatlarini ham ko'rsatkich toifasi sifatida qo'llash imkoniyatini beradi. Masalan,

```
var mon:array[mart,apr, may] of char;  
color:array[red,blue,yellow, black] of integer;
```

Ko'rsatkich(indeks) va ko'rsatkich toifasi tushunchalari o'rtasida farq mavjud bo'lib, ko'rsatkich toifasi massiv elementlari soni va ularning tartiblanganligini bildiradi va u massivni tavsiflash bo'limida ishlatiladi, indeks esa massiv elementining tartib raqamini belgilaydi va operatorlar bo'limidagina ishlatiladi. Agar biror-bir massivga murojat qilish uchun ham to'liq nomi, ya'ni muntazam toifaga mansub o'zgaruvchining nomi ishlatilsa, massivning alohida elementiga murojat qilish uchun ko'rsatkichli o'zgaruvchi ishlatiladi. Masalan,

**A(1)** massiv uchun **A** o'zgaruvchi to'liq o'zgaruvchi (massiv nomi), **A[1]-** ko'rsatkichli o'zgaruvchi bo'lib, u **A** massivning 1-elementini ifodalaydi.



Massiv elementlarining operatorlar bo'limida ishlatilishiga doir misollar:

```
B[5]:=B[3]+1;  
sum:=sum-round(c[k]);  
p1:=sqrt(a[2*i+1]);
```

Delphi dasturlash tilida massivlarni o'zgaruvchilar bo'limida tavsiflashdan tashqari, **TYPE** toifalar bo'limida ham tavsiflash mumkin. Buning uchun **TYPE** bo'limida massiv toifasi nomi va massiv toifasi beriladi, **Var** bo'limida shu toifaga mansub o'zgaruvchilar sanab o'tiladi.

**TYPE** bo'limida toifalarni tavsiflash dasturlashda yaxshi uslub sanaladi va dasturning mantiqiy mukammalligini oshirish imkonini beradi. Massivni **TYPE** bo'limida tavsiflash quyidagi ko'rinishga ega bo'ladi:

```
TYPE <toifa nomi>=array [<ko'rsatkich toifasi>] of <element toifasi>;  
var <o'zgaruvchi yoki massiv nomi>:<toifa nomi>;
```

Yuqorida aytib o'tilgan ma'lumotlarga doir bir nechta misollar ko'rib o'tamiz.

### **1-misol.** Guruhlar ro'yxatini chiqaruvchi dastur tuzamiz.

Dastur kodi va oynani umumiy ko'rinishini keltiramiz:

```
unit unit1;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Variants, Classes, Graphics,
```

```
Controls, Forms, Dialogs, StdCtrls;
```

```
type
```

```
TForm1 = class(Tform)
```

```
Button1: Tbutton;
```

```
Label1: Tlabel;
```

```
procedure Button1Click(Sender: TObject);
```

```
private
```

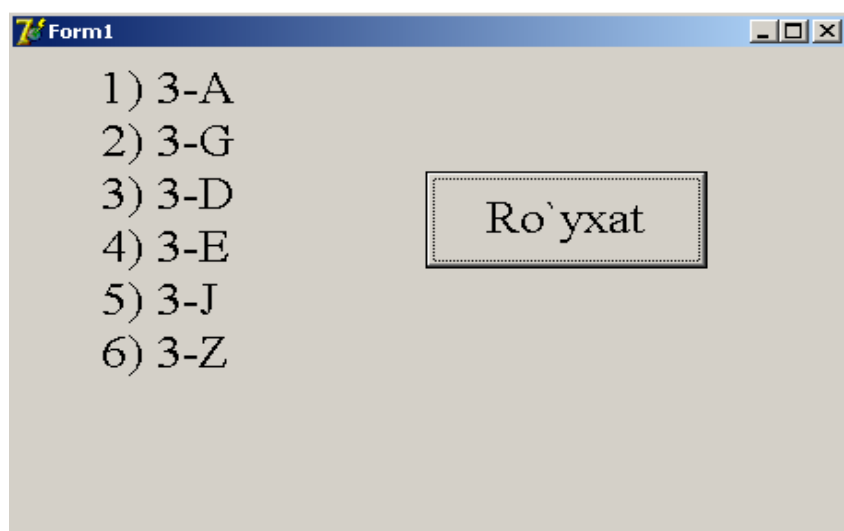
```
{ Private declarations } public
```

```
{ Public declarations } end;
```

```

var
Form1: TForm1;
implementation
{$R *.dfm}
const
NT = 6;
var
Massiv: array[1..NT] of string[10] =
('3-A','3-G','3-D','3-E','3-J','3-Z');
procedure TForm1.Button1Click(Sender: TObject);
var
st:string; // massiv ro'yxati
i:integer; // indeks, massiv elementlarining nomeri
begin
// formallashgan ro'yxat formada joylashish bo'yicha
for i:=1 to NT do
st:= st + IntToStr(i)+ ' ) '+ Massiv[i] + #13;
Label1.Caption:= st; // ro'yxatni chiqarish
end;
end.

```



**1-rasm.**

2-misol.  $A(5)$  massiv elementlari yig'indisi va o'rta arifmetigini toppish.

### Dasturni tuzishning 1-usul.

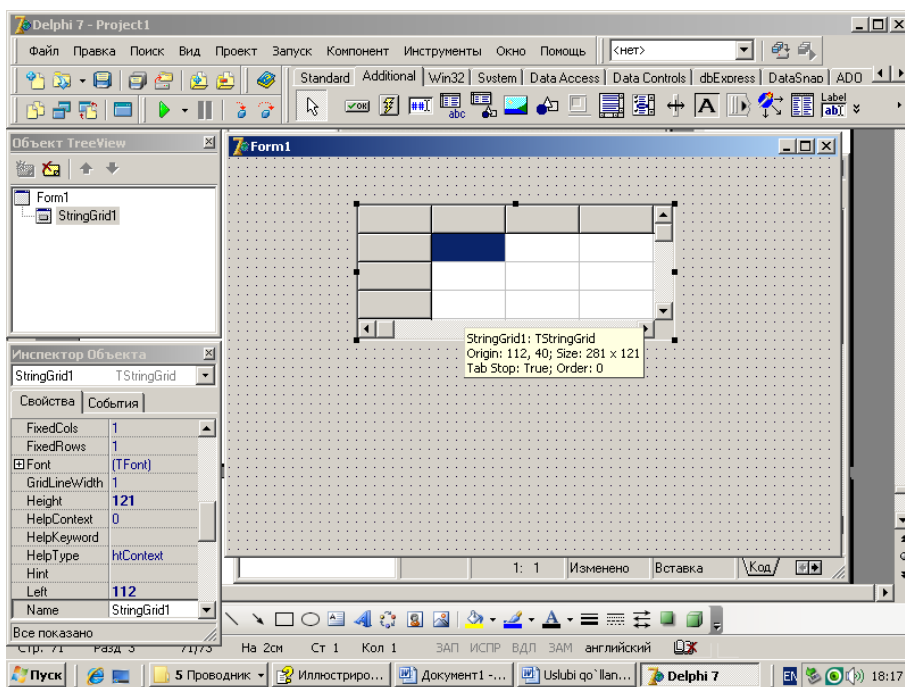
#### 1-bosqich.

1. Komponentalar ro'yxatini Additional bandidan StringGrid1(abc) tugmani tanlaymiz ko'rsatilgan.



2-rasm.

Tugma tanlangandan so'ng, oynaning ko'rinish holati quyidagicha bo'ladi:



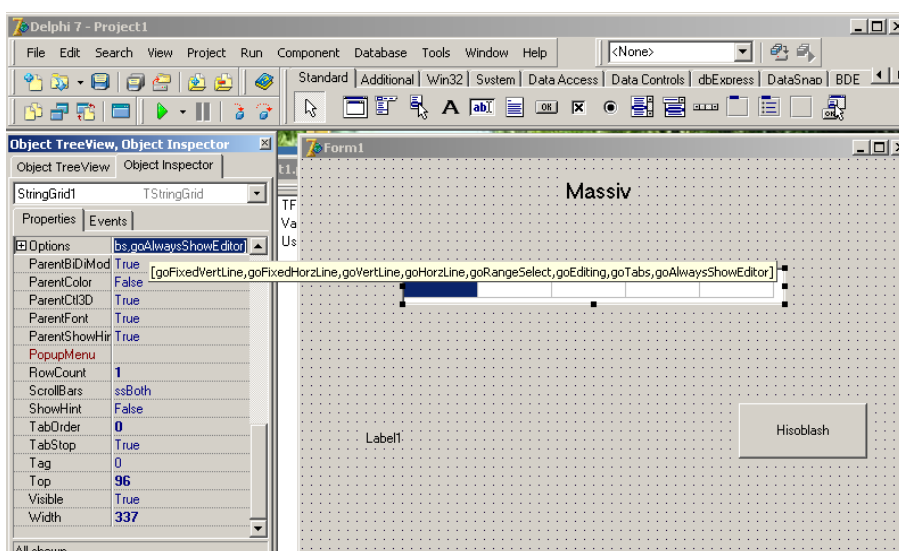
3-rasm.

Hosil bo'lgan jadvaldan 5 ta ustun, 1 ta satr yarating. Bu jadvalni yaratish uchun parametrlar va xossalar bo'limidan **ColCount** hodisalar bandiga 5 raqamini kiritamiz. Satr bo'yicha o'zgartirish kiritish uchun, **RowCount** bandiga 1 raqamini kiritamiz.

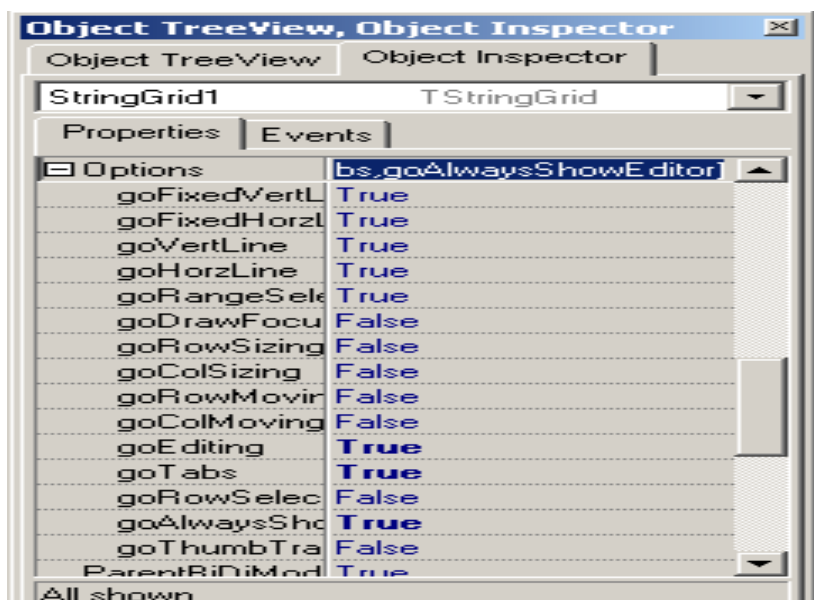
Umumiy holda oynani ishga tayyorlashimiz uchun parametrlar va xossalar bo'limidan jadvalda ko'rsatilgan buyruqlarning tarkibi quyidagicha:

Parametrlar va xossalar bo'limi	Hodisalar bo'limi
ColCount	5
FixedCols	0
RowCount	1
DefaultRowHeight	24
Height	24
DefaultColWidth	64
Width	328
Options . goEditing	True
Options . AlwaysShowEditing	True
Options .goTabs	True

Jadvalda ko'rsatilgan 3 ta **Options.goEditing**, **Options.AlwaysShowEditing**, **Options.goTabs** buyruqlarni hosil qilishimiz uchun parametrlar va xossalar bo'limidan "Options" parametrining "+" ishorasini "-" ishoraga o'tkazamiz (jadval ustiga sichqoncha tugmasini bir marta bosganimizdan so'ng, aytilgan topshiriqni bajarishimiz mumkin).



4-rasm.



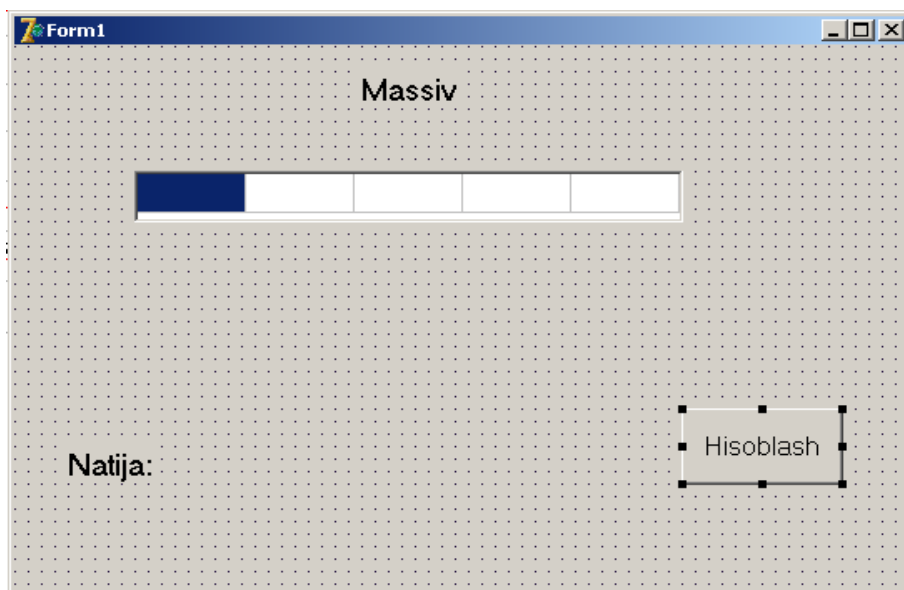
5-rasm.

Hodisalar bo'limining **goEditing** bandida 2 ta **False** va **True** buyrug'i mavjud. **True** buyrug'ini tanlasangiz katakchaga ma'lumot kiritgandan so'ng dastur bilan bog'lanish hosil qiladi aks holda ya'ni **False** buyrug'i tanlasangiz kiritilgan ma'lumot dastur bilan bog'lashga ruxsat etmaydi.

Parametrlar va xossalar bo'limidan "Options" parametirini "+" ishorasini "-" ishoraga o'tkazganimizdan so'ng oynani ko'rinishi quyidagicha bo'ladi:

2. "Label1" tugmasidan foydalanib "Massiv" so'zini kiritamiz.
3. "Label2" tugmasiga "Natija" so'zini kiritamiz.

Natijada oynaning umumiy ko'rinishi quyidagicha bo'ladi:



6-rasm.

## 2-bosqich.

1. Dasturlash maydoniga quyidagi dasturni kiritamiz:

```
unit Unit1;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,  
Dialogs, StdCtrls, Grids;
```

```
Type
```

```
TForm1 = class(TForm)
```

```
StringGrid1: TStringGrid;
```

```
Label1: TLabel;
```

```
Button1: TButton;
```

```
Label2: TLabel;
```

```
procedure Button1Click(Sender: TObject);
```

```
procedure StringGrid1Click(Sender: TObject);
```

```
private
```

```
{ Private declarations }
```

```
public
```

```
{ Public declarations }
```

```
end;
```

```
var
```

```
Form1: TForm1;
```

```
implementation
```

```
{$R *.dfm}
```

```
procedure TForm1.Button1Click(Sender: TObject);
```

```
var
```

```
a : array[1..5] of integer;
```

```
summ: integer;
```

```
sr: real;
```

```

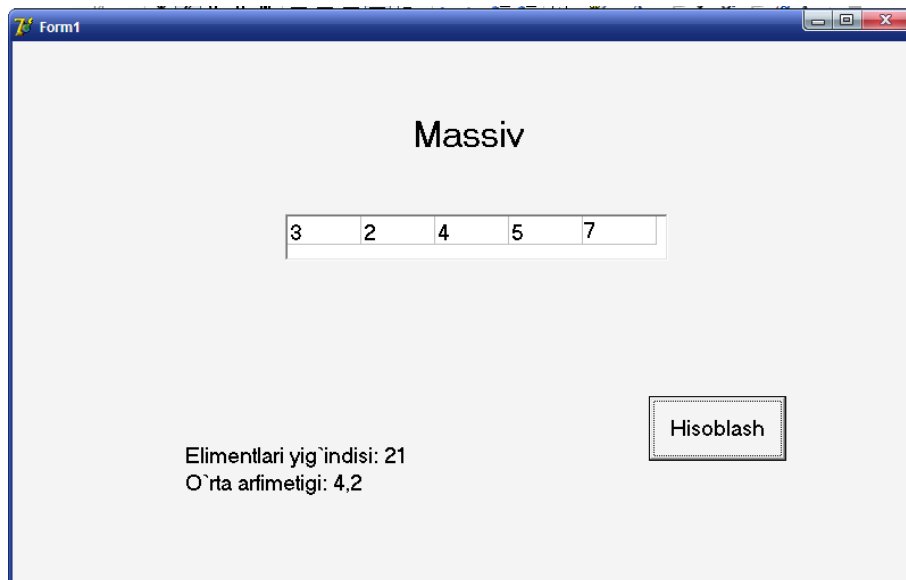
i: integer;
begin
for i:= 1 to 5 do
if Length(StringGrid1.Cells[i-1, 0]) <>0
then a[i] := StrToInt(StringGrid1.Cells[i-1,0])
else a[i] := 0;
summ := 0;
for i :=1 to 5 do
summ := summ + a[i]; sr := summ / 5;
Label1.Caption :=
'Elimentlari yig'indisi: ' + IntToStr(summ)
+ #13+ 'O'rta arfimetigi: ' + FloatToStr(sr);
end;
procedure TForm1.StringGrid1Click(Sender: TObject);
var Key: Char;
begin
case Key of
#8,'0'..'9' : ;
#13:
if StringGrid1.Col < StringGrid1.ColCount-1
then StringGrid1.Col := StringGrid1.Col + 1;
else key := Chr(0);
end;
end;
end.

```

2.Dastur kiritilganidan so'ng, biron bir nom bilan saqlaymiz.

3.Dastur ishini yakunlab ma'lum bir qiymatlarini yachyekalarga kiritamiz

Oynani umumiy ko'rinishi keltiramiz:



7-rasm.

## 2-usul.

### 1-masalani ikkinchi usul bilan hisoblaymiz.

1. **Standart** komponentalar palitrasidan 2 ta **button**, 1 ta **memo** va 1 ta **ComboBox** tugmalarini hosil qilamiz,
2. sichqoncha tumachasini forma oynasiga tez-tez ikki marata bosamiz va dasturni yuqori qismidagi

**var**

**Form1: TForm1;**

Yozuvning pastiga

*a:Array[1..5] of integer;*

*i:integer;*

*s,s1,s2:real;*

yozuvlarni kiritamiz.

3. **"F12"** tugmasini bosgan holda forma oynasiga o'tamiz.

4. Sichqoncha tumachasini forma oynasiga tez-tez ikki marata bosamiz va quyidagi:

*procedure TForm1.FormCreate(Sender: TObject);*

*begin*

*i:=0;*



```
ComboBox1.Clear;
```

```
end;
```

dasturni kiritamiz.

2. **“Button1”** tez-tez ikki marta bosamiz va dasturlash maydoniga

```
begin
```

```
ComboBox1.Items.Add(ComboBox1.text);
```

```
  i:=i+1;
```

```
  a[i]:=StrToInt(ComboBox1.text);
```

```
  ComboBox1.SetFocus;
```

```
end;
```

dasturni kiritamiz.

3. **“Button2”** tez-tez ikki marta bosamiz va dasturlash maydoniga

```
begin
```

```
  S:=0;
```

```
  For i:=1 to 5 do
```

```
    s:=s+a[i];
```

```
  Memo1.Clear;
```

```
  Memo1.Lines.add('Yig'indi =' + floattostr(s));
```

```
  for i:=1 to 5 do
```

```
    s1:=s/5;
```

```
  Memo1.Lines.add('O'rtacha =' + floattostr(s1));
```

```
end;
```

```
end.
```

dasturni kiritamiz.

Dastur kodi va oynani umumiy ko‘rinishini keltiramiz:

```
unit Unit1;
```

```
interface
```

```
uses
```

```
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
```

```

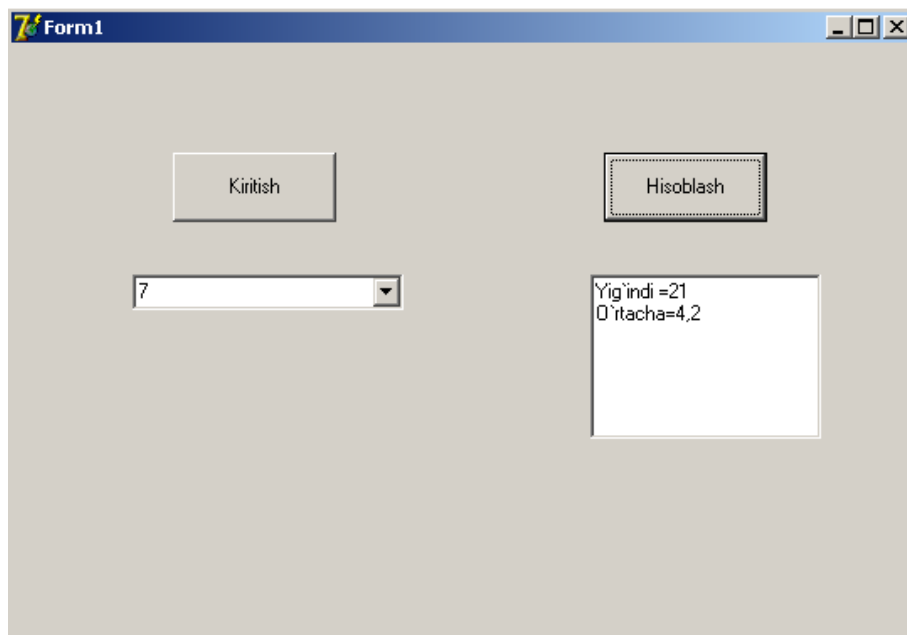
Dialogs, StdCtrls;
type
  TForm1 = class(TForm)
    ComboBox1: TComboBox;
    Button1: TButton;
    Button2: TButton;
    Memo1: TMemo;
    procedure FormCreate(Sender: TObject);
    procedure Button1Click(Sender: TObject);
    procedure Button2Click(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;
var
  Form1: TForm1;
  a:Array[1..5] of integer;
  k,i,maxx,minn:integer;
  s,s1,s2:real;
implementation
{$R *.dfm}
procedure TForm1.FormCreate(Sender: TObject);
begin
  i:=0;
  ComboBox1.Clear;
end;
procedure TForm1.Button1Click(Sender: TObject);
begin

```

```

ComboBox1.Items.Add(ComboBox1.text);
    i:=i+1;
a[i]:=StrToInt(ComboBox1.text);
    ComboBox1.SetFocus;
end;
procedure TForm1.Button2Click(Sender: TObject);
begin
    S:=0;
    For i:=1 to 5 do
        s:=s+a[i];
    Memo1.Clear;
    Memo1.Lines.add('Yig'indi =' + floattostr(s));
    for i:=1 to 5 do
        s1:=s/5;
        Memo1.Lines.add('O'rtacha=' + floattostr(s1));
    end;
end.

```



**8-rasm.**

### **3-usul.**

#### **1-masalani uchunchi usul bilan hisoblaymiz.**

1. **Standart** komponentalar palitrasidan 2 ta **button**, 1 ta **memo**, 1 ta **edit**, va 1 ta **ListBox** tugmalarini hosil qilamiz,

2. sichqoncha tumachasini forma oynasiga tez-tez ikki marata bosamiz va dasturni yuqori qismidagi

**Var Form1: TForm1;**

Yozuvning pastiga

*a:Array[1..5] of integer;*

*i:integer;*

*key,s,s1:real;*

yozuvlarni kiritamiz.

3. **"F12"** tugmasini bosgan holda forma oynasiga o'tamiz.

4. Sichqoncha tumachasini forma oynasiga tez-tez ikki marata bosamiz va quyidagi:

*procedure TForm1.FormCreate(Sender: TObject);*

*begin*

*i:=0;*

*ListBox1.Clear;*

*end;*

dasturni kiritamiz.

4. **"Button1"** tez-tez ikki marta bosamiz va dasturlash maydoniga

*begin*

*ListBox1.Items.Add(Edit1.text);*

*i:=i+1;*

*a[i]:=StrToInt(Edit1.text);*

*Edit1.SetFocus;*

*end;*

dasturni kiritamiz.

## 5. “**Button2**” tez-tez ikki marta bosamiz va dasturlash maydoniga

*begin*

*S:=0;*

*For i:=1 to 5 do*

*s:=s+a[i];*

*Memo1.Clear;*

*Memo1.Lines.add('Yig'indi =' + floattostr(s));*

*for i:=1 to 5 do*

*s1:=s/5;*

*Memo1.Lines.add('O'rtacha=' + floattostr(s1));*

*end;*

*end.*

dasturni kiritamiz.

Dastur kodi va oynani umumiy ko'rinishini keltiramiz:

*unit Unit1;*

*interface*

*uses*

*Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,*

*Dialogs, StdCtrls;*

*type*

*TForm1 = class(TForm)*

*Edit1: TEdit;*

*ListBox1: TListBox;*

*Button1: TButton;*

*Button2: TButton;*

*Memo1: TMemo;*

*procedure FormCreate(Sender: TObject);*

*procedure Button1Click(Sender: TObject);*

*procedure Button2Click(Sender: TObject);*

```

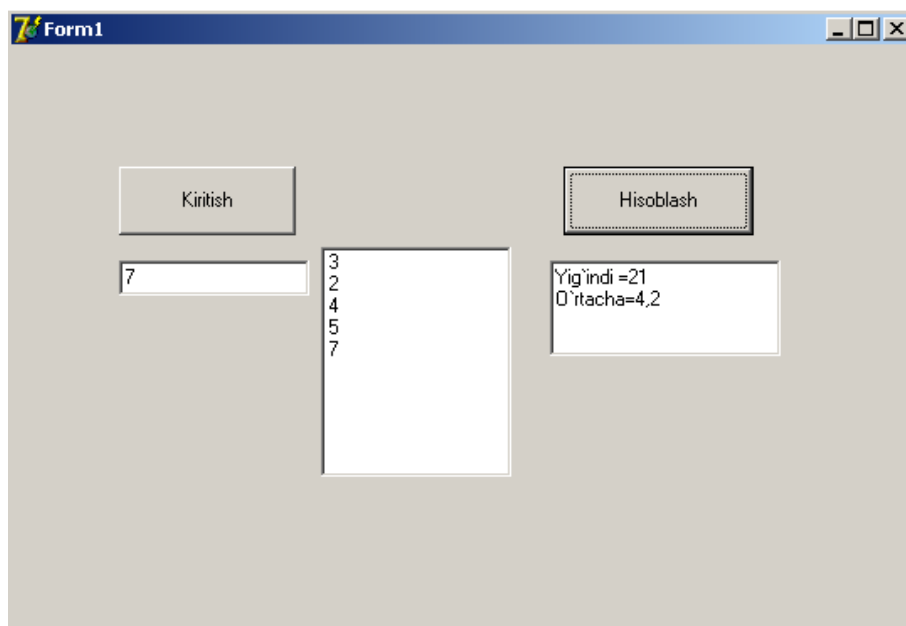
    procedure Edit1Change(Sender: TObject);
private
    { Private declarations }
public
    { Public declarations }
end;
var
    Form1: TForm1;
    a:Array[1..5] of integer;
    k,i,maxx,minn:integer;
    key,s,s1,s2:real;
implementation
{$R *.dfm}
procedure TForm1.FormCreate(Sender: TObject);
begin
    i:=0;
    ListBox1.Clear;
end;
procedure TForm1.Button1Click(Sender: TObject);
begin
    ListBox1.Items.Add(Edit1.text);
        i:=i+1;
        a[i]:=StrToInt(Edit1.text);
        Edit1.SetFocus;
end;
procedure TForm1.Button2Click(Sender: TObject);
begin
    S:=0;
    For i:=1 to 5 do

```

```

s:=s+a[i];
Memo1.Clear;
Memo1.Lines.add('Yig'indi =' + floattostr(s));
for i:=1 to 5 do
s1:=s/5;
Memo1.Lines.add('O'rtacha=' + floattostr(s1));
end;
procedure TForm1.Edit1Change(Sender: TObject);
begin
If key=13 Then Button1.SetFocus;
end;
end.

```



**9-rasm.**

**3-misol.** A(N) vektor elementlari ko`paytmasini hisoblash dasturi.

Dastur kodi va oynani umumiy ko`rinishini keltiramiz:

```
unit Unit1;
```

```
interface
```

```
uses
```

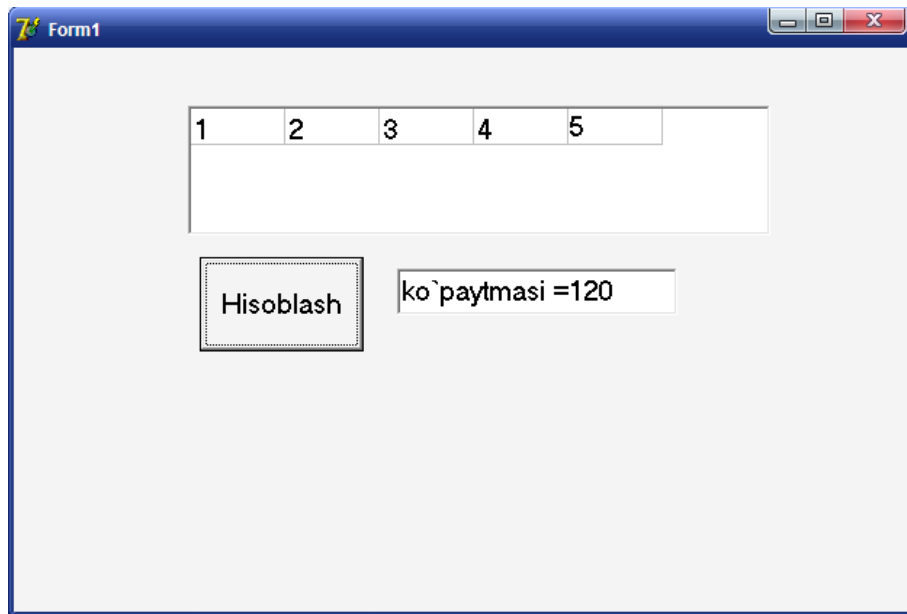
*Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,*

```

Dialogs, StdCtrls, Grids;
type
  TForm1 = class(TForm)
    Button1: TButton;
    StringGrid1: TStringGrid;
    Edit1: TEdit;
    procedure Button1Click(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;
var
  Form1: TForm1;
implementation
  {$R *.dfm}
  procedure TForm1.Button1Click(Sender: TObject);
  const N=5;
  var a,b:array[1..N] of integer;
  p,i:integer;
  begin
  for i:=1 to n do
  a[i]:=StrToInt(StringGrid1.Cells[i-1,0]);
  p:=1;
  for i:=1 to n do
  p:=p*a[i];
  edit1.Text:='ko`paytmasi =' + FloatToStr(p);
  end;
end.

```





**10-rasm.**

**4-misol.** A(N) vektorni manfiy elementlarni topish dasturi.

Dastur kodi va oynani umumiy ko`rinishini keltiramiz:

*unit Unit1;*

*interface*

*uses*

*Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,  
Dialogs, StdCtrls, Grids;*

*type*

*TForm1 = class(TForm)*

*Button1: TButton;*

*StringGrid1: TStringGrid;*

*StringGrid2: TStringGrid;*

*procedure Button1Click(Sender: TObject);*

*private*

*{ Private declarations }*

*public*

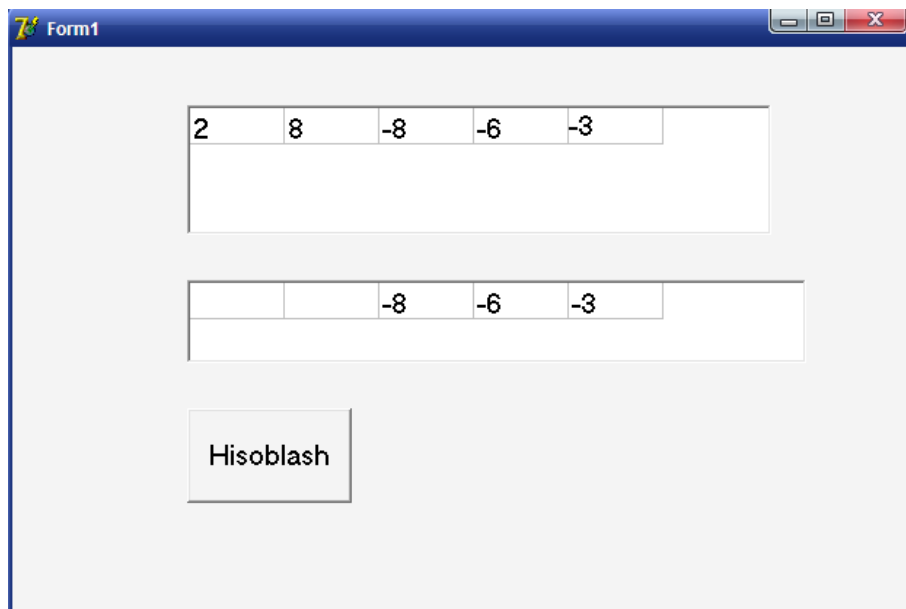
*{ Public declarations }*

*end;*

```

var
  Form1: TForm1;
implementation
{$R *.dfm}
procedure TForm1.Button1Click(Sender: TObject);
const N=5;
var a,b:array[1..N] of integer;
min,p,i:integer;
begin
for i:=1 to n do
  begin a[i]:=StrToInt(StringGrid1.Cells[i-1,0]); end;
for i:=1 to n do
  begin
if a[i]<0 then begin b[i]:=a[i];
stringgrid2.Cells[i-1,0]:=IntToStr(b[i]); end;
end;
end; end.

```



**11-rasm.**

**5-misol.** A(5) vektornining eng kichigini va element nomerini topish dasturi.

Dastur kodi va oynani umumiy ko'rinishini keltiramiz:

*Unit1;*

*interface*

*uses*

*Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms, Dialogs, StdCtrls, Grids;*

*type*

*TForm1 = class(TForm)*

*StringGrid1: TStringGrid;*

*Label1: TLabel;*

*Label2: TLabel;*

*Button1: TButton;*

*procedure Button1Click(Sender: TObject);*

*procedure StringGrid1Click(Sender: TObject);*

*private*

*{ Private declarations }*

*public*

*{ Public declarations }*

*end;*

*var*

*Form1: TForm1;*

*implementation*

*{\$R \*.dfm}*

*procedure TForm1.Button1Click(Sender: TObject);*

*const*

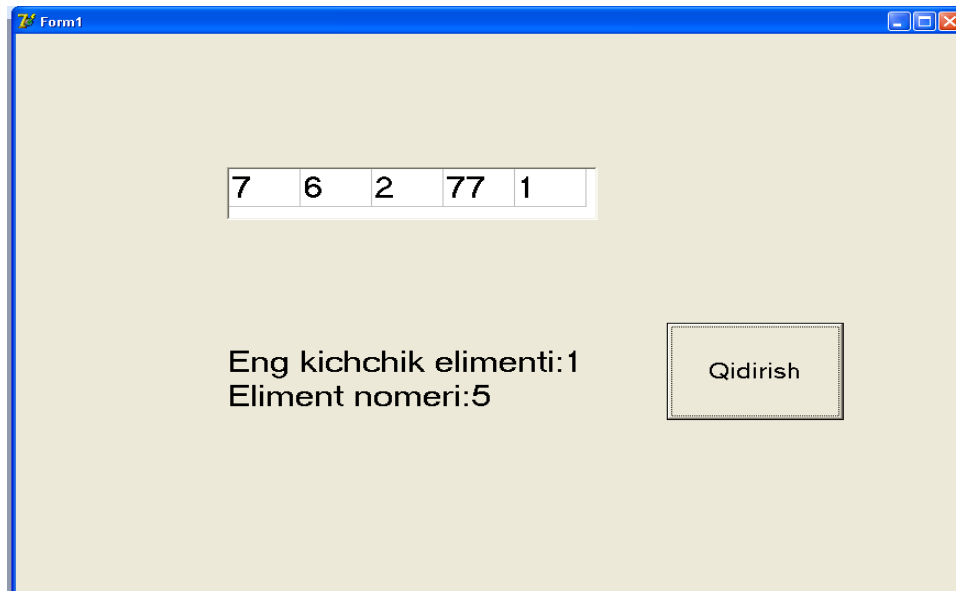
*k=5;*

*var*

```

a:array[1..k]of integer;
min:integer;
i:integer;
begin
  for i:=1 to k do
    a[i]:=StrToInt(StringGrid1.Cells[i-1,0]);
    min:=1;
    for i:=2 to k do
      if a[i]< a[min]then min:=i;
    label2.caption:='Eng kichchik elimenti:'
    +IntToStr(a[min])+#13+'Eliment nomeri:'+ IntToStr(min);
  end;
  procedure TForm1.StringGrid1Click(Sender: TObject);
  var Key: Char;
  begin
    case Key of
      #8,'0'..'9': ;
      #13:
        if StringGrid1.Col < StringGrid1.ColCount-1
        then StringGrid1.Col:= StringGrid1.Col + 1;
        else key := Chr(0);
    end;
  end;
end.

```



**12-rasm.**

**6-misol.** A(10) vektorning toq va juft o`rinlarida joylashgan elementlarini B(K) vektorga ketma-ket yozish dasturi.

Dastur kodi va oynani umumiy ko`rinishini keltiramiz:

*unit Unit1;*

*interface*

*uses*

*Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,  
Dialogs, StdCtrls, Buttons, Grids;*

*type*

*TForm1 = class(TForm)*

*StringGrid1: TStringGrid;*

*BitBtn1: TBitBtn;*

*StringGrid2: TStringGrid;*

*GroupBox1: TGroupBox;*

*RadioButton1: TRadioButton;*

*RadioButton2: TRadioButton;*

*procedure BitBtn1Click(Sender: TObject);*

*private*

```

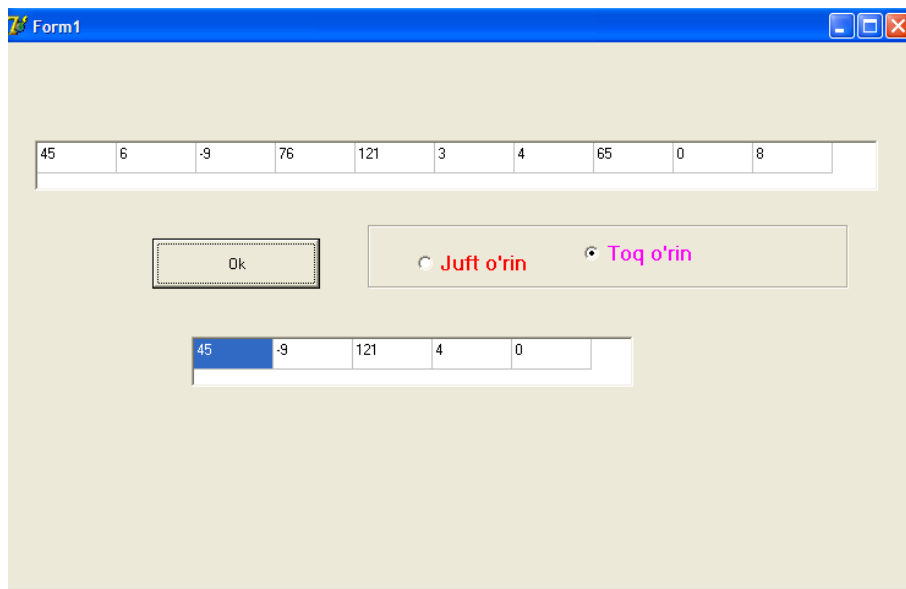
    { Private declarations }
public
    { Public declarations }
end;
var
    Form1: TForm1;
implementation
{$R *.dfm}
procedure TForm1.BitBtn1Click(Sender: TObject);
var i,k:integer;
begin
k:=0;
if RadioButton1.Checked then
begin
for i:=0 to 9 do
begin
if odd(i) then
begin
StringGrid2.Cells[k,0]:=StringGrid1.Cells[i,0];
k:=k+1;
end;
end;
end
else
begin
for i:=0 to 9 do
begin
if not odd(i) then
begin
StringGrid2.Cells[k,0]:=StringGrid1.Cells[i,0];

```

```

    k:=k+1;
end;
end;
end;
end;
end.

```



**13-rasm.**

**7-misol.** X(12) vektor elementlarini kamayib borish tartibda joylashtirish dasturi.

Dastur kodi va oynani umumiy ko'rinishini keltiramiz:

```

unit Unit1;
interface
uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, StdCtrls, Grids;
type
  TForm1 = class(TForm)
    StringGrid1: TStringGrid;
    StringGrid2: TStringGrid;
    Button1: TButton;
  procedure Button1Click(Sender: TObject);

```

```

private
  { Private declarations }
public
  { Public declarations }
end;
var
  Form1: TForm1;
implementation
{$R *.dfm}
procedure TForm1.Button1Click(Sender: TObject);
var i,k:integer;
    x:array[1..13] of integer;
    min:integer;
begin
  for i:=1 to 12 do
    begin
      for k:=1 to 12 do
        begin
          x[i]:=StrToInt(StringGrid1.Cells[0,i]);
        end;
      end;
    min:=x[1];
    for i:=1 to 11 do
      begin
        for k:=i+1 to 12 do
          begin
            if x[i]<x[k] then
              begin
                min:=x[i];
                x[i]:=x[k];
              end;
            end;
          end;
        end;
      end;
    end;
  end;
end;

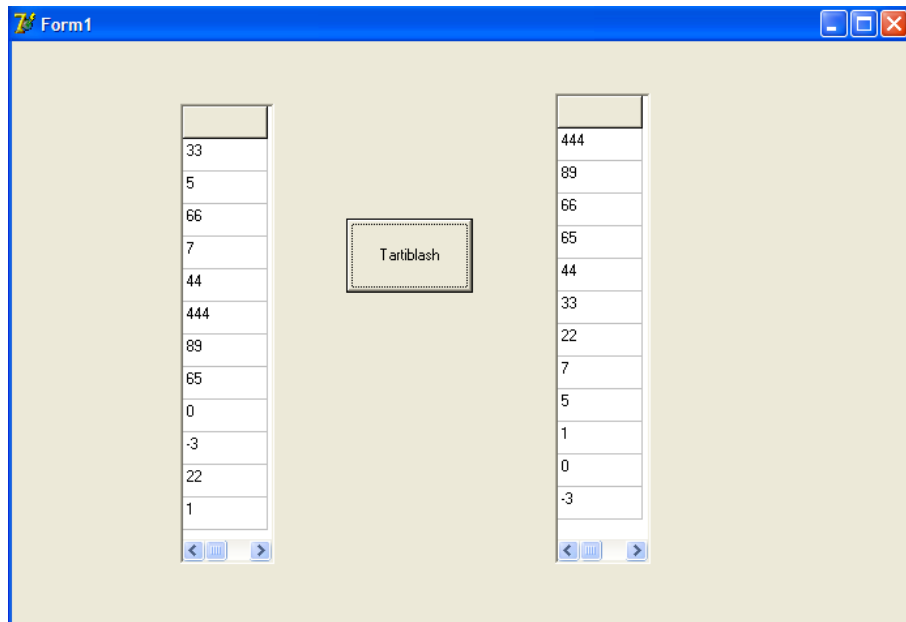
```



```

        x[k]:=min;
    end;
end;
end;
for i:=1 to 12 do StringGrid2.Cells[0,i]:=IntToStr(x[i]);
end;
end.

```



**14-rasm.**

**8-misol.** A(15) vektorning eng katta elementi o`rnini eng kichik elementi o`rniga almashtirish dasturi.

Dastur kodi va oynani umumiy ko`rinishini keltiramiz:

*unit Unit1;*

*interface*

*uses*

*Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,  
Dialogs, StdCtrls, Grids;*

*type*

*TForm1 = class(TForm)*

*StringGrid1: TStringGrid;*

```

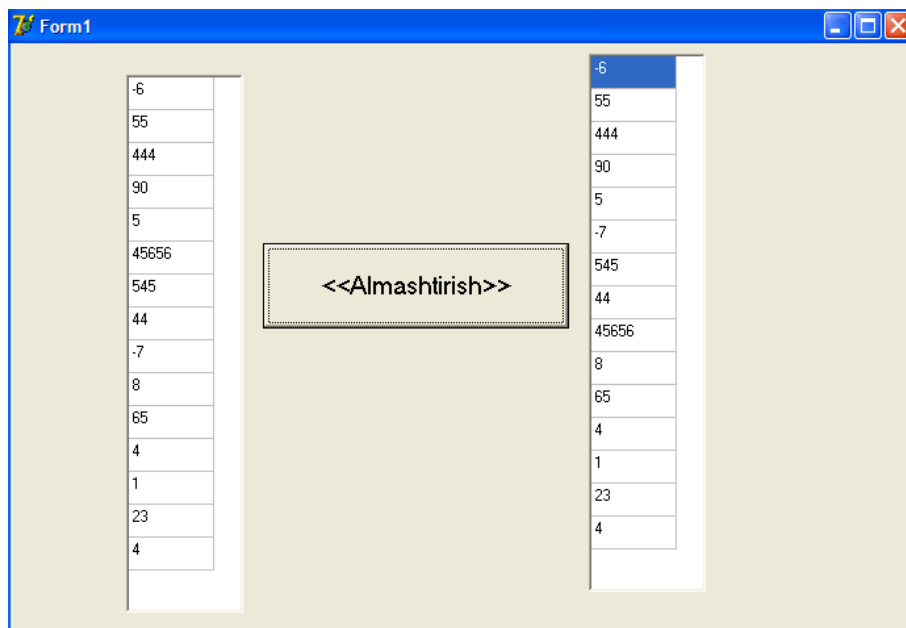
Button1: TButton;
StringGrid2: TStringGrid;
procedure Button1Click(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;
var
  Form1: TForm1;
implementation
  {$R *.dfm}
  procedure TForm1.Button1Click(Sender: TObject);
  Var i:integer;
  a:array[0..14] of integer;
  alm,max,min,maxid,minid:integer;
begin
  for i:=0 to 14 do
    begin
      a[i]:=strtoint(StringGrid1.Cells[0,i]);
    end;
  min:=a[0]; max:=a[0];
  for i:=0 to 14 do
    begin
      if min>a[i] then
        begin
          min:=a[i];
          minid:=i;
        end;
      if max<a[i] then

```

```

begin
max:=a[i];
maxid:=i;
end;
end;
alm:=a[maxid];
a[maxid]:=a[minid];
a[minid]:=alm;
for i:=0 to 14 do
begin
StringGrid2.Cells[0,i]:=IntToStr(a[i]);
end;
end;
end.

```



**16-rasm.**

## Mustaqil bajarish uchun topshiriqlar

1.  $y = \frac{\sum_{j=1}^6 \ln|x_i + 2,5|}{\prod_{i=1}^6 x_i^2}$  ni hisoblash dasturini tuzing.
2.  $y = \sum_{i=1}^5 \frac{ilz_i}{2^i}$  ni hisoblash dasturini tuzing.
3.  $A$  10 vektor elementlarini eng kichigini topish dasturini tuzing.
4.  $A$  10 vektor elementlarini eng kattasini topish dasturini tuzing.
5. Uchunchi tartibli kvadrat matritsaning teskarisini topish dasturini tuzing.
6.  $A(N,N)$  massivning  $p$  va  $q$  tartib raqamli satrlari o'rnini almashtirovchi dasturini tuzing.
7.  $X(K,L)$  massivning eng katta va eng kichik elementlari o'rnini almashtiruvchi dastur tuzing.
8.  $A(3,4)$  massivning satr elementlari ko'paytmasidan  $B$  massivni hosil qilish dasturini tuzing.
9.  $Z(3,4)$  massivning har bir ustunidagi manfiy elementlar sonidan tashkil topgan  $M$  vektorni hosil qilish dasturini tuzing.
10.  $A(M,N)$  massiv berilgan. Har bir satrdagi eng kichik elementlar orasidan eng kattasini va u joylashgan tartib raqamini aniqlash dasturini tuzing.
11. Diagonal elementlaridan tashqari barchga elementlari nolga teng bo'lgan  $C(M,M)$  massiv tashkil etuvchi dastur tuzing.
12.  $K(3,4)$  massivning musbat elementlaridan tashkil topgan  $L$  vektorni hosil qiluvchi dastur tuzing.
13.  $A(10,15)$  massivning har bir ustunidagi musbat elementlarining sonini va yig'ingisini eslab qoluvchi va hisoblovchi dastur tuzing. Natija ikkita satr ko'rinishida chop etilsin.  $a_{ij} > 0$ .

14.  $N(8,6)$  massivning har bir satridagi manfiy elementlar sonini va ularning yig'indisini aniqlovchi va bu yig'indini massiv ko'rinishda chop etuvchi dastur tuzing.
15.  $S(10,20)$  massivning eng katta elementini toping va matritsaning har bir elementini o'nga bo'lib, hosil bo'lgan massivni chop etish dasturini tuzing.
16.  $A(4,3)$  massivning manfiy elementlarini nisbatini hisoblash dasturini tuzing.
17.  $A(3,3)$  massivning har bir satrida joylashgan elementlarining eng kattasini topib uni  $B(1)$  massivga chiqarish dasturini tuzing.
18.  $A(10,10)$  massivning satr elementlarini eng kattasini chiqarish dasturini tuzing.
19.  $A(4,4)$  massivning satr va ustun elementlari yig'indisini kichigini topish dasturini tuzing.
20.  $A(4,4)$  massivning eng katta elementi va uning turgan o'rnini aniqlash dasturini tuzing.
21.  $A(5,5)$  massivning ustun elementlarini satr shaklda chiqarish dasturini tuzing.
22.  $c_{ij} = a_{ij} - b_{ij}$  hisoblash dasturini tuzing. Bunda  $i = \overline{1,4}, j = \overline{1,4}$
23.  $c_{ij} = a_{ij} * b_{ij}$  hisoblash dasturini tuzing. Bunda  $i = \overline{1,4}, j = \overline{1,4}$
24.  $A(5,5)$  massivning har bir satr elementlarini o'rta arifmetigini topish dasturini tuzing.

## FOYDALANILGAN ADABIYOTLAR

1. Sh.A.Nazirov, M.M.Musayev, A.Ne`matov, R.V.Qobulov. Delphi tilida dasturlash asoslari". Toshkent. 2007 yil.
2. A. Ahmedov, N.Tayloqov. Informatika. Toshkent, "O'zbekiston". 2008 -y.
3. P.Karimov, S.Irisqulov, A.Isabayev. Dasturlash. Toshkent, "O'zbekiston". 2003-y.
4. A.A.Xaldjigitov, Sh.F.Madraximov, A.M.Ikramov, S.I.Rasulov. Paskal tilida dasturlash bo'yicha masalalar to'plami. 2002-y.
5. R.A.Ro`ziyev, G`.R.Yodgorov, O`.M.Mirsanov. Delphi dasturlash tilini o`rganish. Uslubiy qo`llanma. Navoiy 2012-y.
6. Поляков Д.В., Круглов И.Ю. Программирование в среде TURBO PASCAL . –М. Изд-во МАИ, 1992.
7. С.А. Немнюгин. Turbo Pascal, Москва Харьков-Минск 2000.
8. Иллюстрированный самоучитель по Delphi 7 для начинающих. Lib.xost.ru Kutubxonasi.
9. Н.Б.Культин. Delphi 6. Программирование на Object Pascal.