

**O'ZBEKISTON RESPUBLIKASI XALQ TA'LIM
VAZIRLIGI**

AJINIYOZ NOMIDAGI NUKUS DAVLAT PEDAGOGIKA INSTITUTI

Fizika-matematika fakulteti

Informatika o`qitish metodikasi kafedrası

4^z-kurs talabasi O.Qunnazarovning

BITIRUV MALAKAVIY ISHI

**Mavzu: Ob`ektga mo`ljallangan dasturlashda tajriba
mashg`ulotlarin o`tkazish metodikasi.**

Ilmiy rahbar:

e.i.k., dots. A. Abdullaev

Kafedra mudiri:

f.-m.i.k. A.Tureniyazova

NUKUS - 2016

Mundarija

Kirish	3
1-bob.Ob`ektga mo`ljallangan dasturlashda tajriba mashg`ulotlarin o`tkazish asoslari	
1.1. Masalaning qo`yilishi	5
1.2. Forma va ob`ektlarni loyihalashtirish	6
1.3. Dastur strukturasi va berilganlar tipi	9
 2-bob. Ma`lumotlar bilan ishlash metodikasi	
2.1. Fayllar ustida amallar bajarish	14
2.2. Delphida ro`yxatlar	23
2.3. Jadvallar ustida amallar	28
 3-bob. Amaliy masalalarga tajriba mashg`ulotlarin o`tkazish metodikasi	
3.1. Funktsiyalar grafigin chizish	35
3.2. Grafik ob`ektlar	39
 Xulosa	48
Adabiyotlar	49

Kirish

Ushbu bitiruv malakaviy ishida “Algortimlashtirish va dasturlash tillari” fanidan tajriba mashg`ulotlarni bajarish uchun ko`rsatmalar, tajriba mashg`ulot variantlari va namunalar keltirilgan. Delphi dasturlash tili muhitida ishlash va uni o`rganish variantlarni tanlab dastur tuzish va dasturni kiritish, natijani qayta ishlash orqali mustahkamlanadi. Bitiruv malakaviy ishining natijalarin umumiy orta maktablar, akademik lisey, kasb-hunar kollejlari va institut o`quvchi va talabalari foydalanishlari mumkin.

Etiboringizga havola etilayotgan ushbu bitiruv malakaviy ishi, hozirgi kunda juda kerakli bo`lgan, zamonaviy tillarning asosi Delphi dasturlash tili haqida, unda ishlash, dastur tuzish, buyruqlar, operatorlar va ularning vazifalari, tajriba ishlarining namunalari, topshiriqlari keltirilgan.

1991 yildagi ma`lumotlarga qaraganda Delphi dasturlash tili o`zining qulayligi, universalligi sababli yer yuzidagi dasturlovchilarning 30 foizi shu dasturlash tilidan foydalangan. Bu esa 100 ortiq dasturlash tillari ichida juda keng foydalanuvchiga ega ekanligidan dalolat beradi. Ma`lumki hozirgi kunda bu dasturlash tizimini zamonaviy verisyasi visual dasturlash muhiti Borland Delphi ni o`rganishga bag`ishlangan rus va xorijiy tillarda 100 dan ortiq adabiyotlar mavjud. O`zbek titlida esa bu dasturlashga oid qo`llanma barmoq bilan sanash darajasida. Chop etilganlari ham kam nusxada bo`lib foydalanuvchilar momasini qondirmaydi.

Yuqorida aytilganlarni hisobga olib ushbu bitiruv malakaviy ishining maqsadin qo`yidagshicha ifodalaymiz.

Ishning maqsadi. Bitiruv malakaviy ishida Ob`ektga mo`ljallangan dasturlashda tajriba mashg`ulotlarin o`tkazish metodikasi qaralgan bo`lib, unda Delphi-7 dasturlash tilida berilgan topshiriqlar uchun foydalanuvchilarga tajriba mashg`ulotlarin bajarish tartibin o`rgatish hisoblanadi.

**Tadqiqot ob`ekti.** Topshiriqlar, Delphi-7 da dasturlash texnologiyasi, tajriba mashg`ulotlarin o`tkazish texnologiyasi, ob`ektga mo`ljallangan algoritmlashtirish va dasturlash tillari yotadi.

Tadqiqot metodikasi. Tadqiqot olib borishda analiz va sintez usuli, davlatimiz tomonidan chiqarilgan ta'lim to'g'risidagi xujjatlardan foydalanildi.

Tadqiqot natijalarining ilmiy tomondan yangilik darajasi. Ishning asosiy yangiligi bo'lib maktab kursidagi masalalarni tajriba asosida echishni ob'ektga mo'ljallangan algoritmlashtirish va dasturlash tillari asosida foydalanuvchilarga o'rgatish va bu soha bo'yicha dasturiy interaktiv tizim yaratish yotadi.

Tadqiqot natijalarining amaliy ahamiyati. Bitiruv malakaviy ishida olingan natijalarni maktab, kollej va akademik litsey talabalari uchun o'quv rejasidagi fanlar bo'yicha masalalarni ob'ektga mo'ljallangan dasturlar tillaridan foydalanib mustaqil yechishni va tajriba o'tkazishda asosiy uslubiyat bo'lib hisoblanadi.

Ishning tuzilishi. Bitiruv malakaviy ishi kirish, uchta bobdan, xulosa, foydalanilgan adabiyotlar va qo'shimchalardan iborat.

Ishning asosiy natijalari va strukturasi. Ushbu bitiruv malakaviy ishida qo'yidagi masalalar qaraladi: I-bob «Ob'ektga mo'ljallangan dasturlashda tajriba mashg'ulotlarin o'tkazish asoslari» atamasida bo'lib: masalaning qo'yilishi, forma va ob'ektlarni loyihalashtirish, dastur strukturasi va berilganlar tipi masalalari qaraladi va 1-bob buyicha xulosa keltirilgan.

II-bob «Ma'lumotlar bilan ishlash metodikasi» atamasida bo'lib fayllar ustida amallar bajarish, Delphi da ro'xatlar, jadvallar ustida amallar qaralada va 2-bob buyicha xulosa chiqarilgan.

III-bob «Amaliy masalalarga tajriba mashg'ulotlarin o'tkazish metodikasi» atamasida bo'lib funktsiyalar grafigin chizish, grafik ob'ektlar bilan ishlash metodikasi qaraladi. 3-bob buyicha xulosa keltirilgan.

Ishning oxirida olingan natijalar asosida umumiy xulosa va takliflar, foydalanilgan adabiyotlar ro'yxati, qo'shimcha ma'lumotlar keltirilgan.

Xulosa va takliflarning qisqacha umumlashtirilgan ko'rinishi. Bitiruv malakaviy ishida ob'ektga mo'ljallangan dasturlash tillarida foydalanuvchilarni tajriba mashg'ulotlarin o'tkazish metodikasi ko'rsatilgan.

1-bob.Ob`ektga mo`ljallangan dasturlashda tajriba mashg`ulotlarini o`tkazish asoslari

1.1. Masalaning qo`yilishi

Ob`ektga mo`ljallangan dasturlash tillari Turbo Pascal, VBA, C++, JAVA, Delphi va boshqa yuqori darajali tillar yotadi. Ma`lumki, har qanday fanni o`rganish fan bo`yicha nazariy ma`lumotlarni o`zlashtirishdan boshlanadi. So`ngra o`rganilgan nazariy ma`lumotlarni amalda qo`llay bilish malakasini xosil qilish uchun amaliy va tajriba mashg`ulotlari o`tkaziladi. Dasturlash asoslarini chuqur o`zlashtirishda ayniqsa tajriba mashg`ulotlarining salmog`i juda kattadir. Bu mashg`ulotlarda talaba mustaqil ravishda berilgan topshiriqlar bo`yicha qo`yilgan masalani yechishning mantiqiy ketma-ketligini ishlab chiqadi va yaratilgan dasturiy ta`minot bo`yicha natijalar olishga hamda ularni taxlil qilishga erishadi. Shuning uchun tajriba mashg`ulotlarini sifatli, yuqori saviyada o`tkazilishi talabalarning fan bo`yicha o`zlashtirish darajasiga ijobiy ta`sir o`tkazadi.

Bitiruv malakaviy ishi « Algoritmlashtirish va dasturlash tillari» fanining bir qismi bo`lgan Delphi-7 dasturlash tili asosida tajriba mashg`ulotlarini o`tkazish bo`yicha zarur bo`lgan barcha yo`riqnomalarni o`z ichiga olgan va Davlat ta`lim standartlariga mos namunaviy dastur hamda unga mos ishchi dasturlar asosida tuzilgan.

Tajriba mashg`ulotlarini bajarish uchun talabalarga quyidagi talablar qo`yiladi:

1. Tajriba ishini bajarish uchun zarur bo`lgan nazariy ma`lumotlarni ko`rsatilgan adabiyotlardan to`plash;
2. Belgilangan variant bo`yicha shaxsiy topshiriqlarni o`z vaqtida olish;
3. Ish rejasida belgilangan masalalarning qo`yilishi, uning mohiyati va amaliy masalalarni yechishdagi ahamiyati xaqida to`liq ma`lumotga ega bo`lish;
4. Masalani hal qilishning ishchi algoritmlarini ishlab chiqish va ularni blok - sxemalar ko`rinishida ifodalash;

5. Ishlab chiqarilgan algoritm bo'yicha Delphi dasturlash tilida dastur yaratish va uni ishga sozlash;
6. Tuzilgan dasturni tekshirib ko'rish va olinayotgan natijalarning to'g'riligiga ishonch bildirish;
7. Shaxsiy topshiriqning dasturidan olingan natijalarni zarur formada chop etishni tashkillash;
8. Bajarilgan ishlarni belgilangan reja asosida rasmiylashtirish va uni o'qituvchi huzurida ximoya qilish (tajriba darsi mobaynida);
9. Ximoya qilingan, o'qituvchi tomonidan ijobiy baholangan tajriba ishi hisobotini o'qituvchiga topshirish;
10. Tajriba ishini o'z vaqtida bajarmagan talaba o'qituvchi belgilagan vaqtda (dars mashg'ulotlaridan bo'sh vaqtda) kafedraga kelib ishni bajaradi va belgilangan tartibda uni ximoya qiladi;
11. Agar talaba dars mobaynida va qo'shimcha belgilangan vaqtda ham tajriba ishini bajarmasa, shu mavzu uchun reyting ballarini olmagan va mavzuni o'zlashtirmagan hisoblanadi.

Tajriba ishlarini bajarish uchun kerakli jihozlar, texnik va dasturiy vositalar:

Kompyuter (Pentium IV);

Operasion tizim (Windows 2000, Windows xp);

Borland Delphi7 dasturlari

Delphi-7 har xil dasturlarni ishlab chiqarish imkoniyatin beradi va tajriba ishini bajarishni ikkita rejimda olib boramiz:

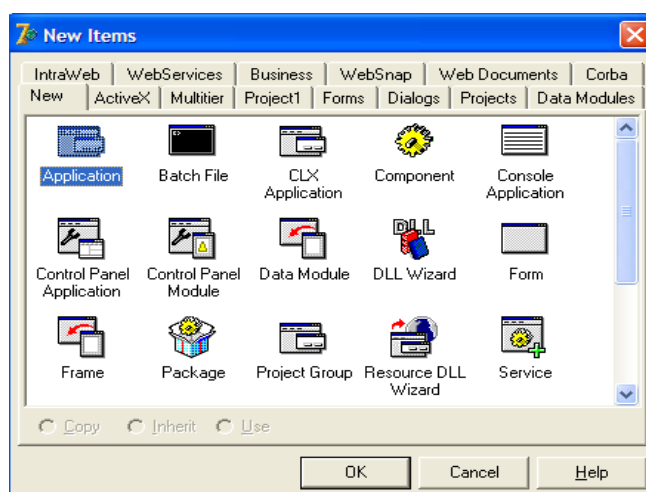
1. Konsolli ilova (**Console Application**) rejimida.
2. Loyiha ilova (**Application**) grafikali interfeysga ega **Windows** operatsion tizimiga moslashgan rejimda.

1.2. Forma va ob'ektlarni loyihalashtirish

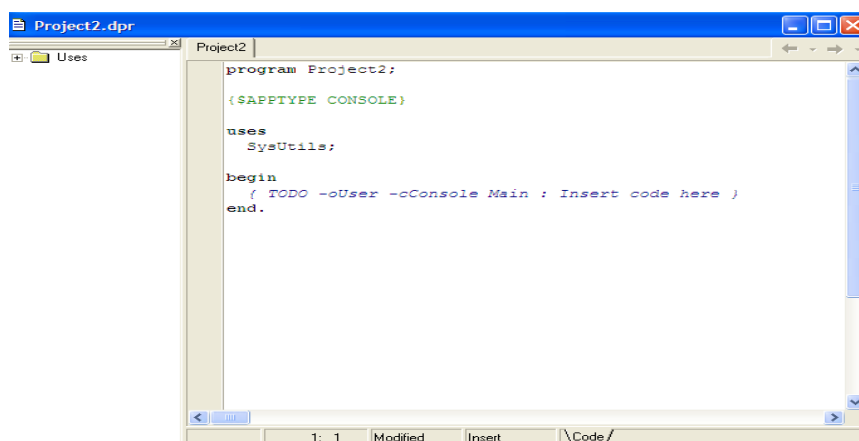
Konsolli ilova (**Console Application**) rejimida ishlash uchun menyu qatoridan File=>New=>Other.. bo'yruqlarin tanlab konsolli rejim yaratamiz va 1-rasmda ko'rsatilgan loyiha turin tanglash oynasin chaqiradi. Undan « **Console**

Application» ni tanlaymiz. Delphi-7 kod tahriri oynasida loyiha uchun 2-rasmda ko`rsatilgan tayyorgarlikni shakllantiradi va Project so`ziga ega bo`ladi. Konsolli rejim formasi (demak o`nga mos keluvchi modul) yo`q. Ob`ektlar inspektori bo`sh, sababi loyihada ob`ektlar yo`q. Loyiha dasturi tayyorgarligida qo`yidagilar o`tmashgan:

- Kompilyatorga konsolli qo`llanmani yaratishni bo`yuruvchi Direktiva {\$APPTYPE CONSOLE}.
- Kommentariyalar // Insert user code here (foydalanuvchi kodin shu erga kiriting).
- Kod provodnigi panelida (2-rasmning chap tarafida) **SysUtils** moduliga yunalish ko`rsatilgan–sistemali utilitalar. Bu modul bazi foydalanuvchi kichik dasturlarini o`z ichiga oladi. U Delphi-7 yunalishni avtomat turda **uses** satrida yaratadi.



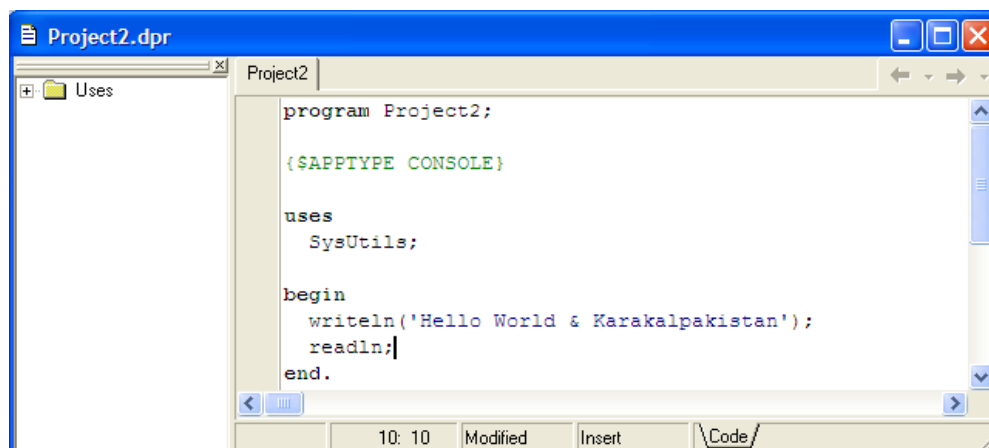
1-rasm.



2-rasm.

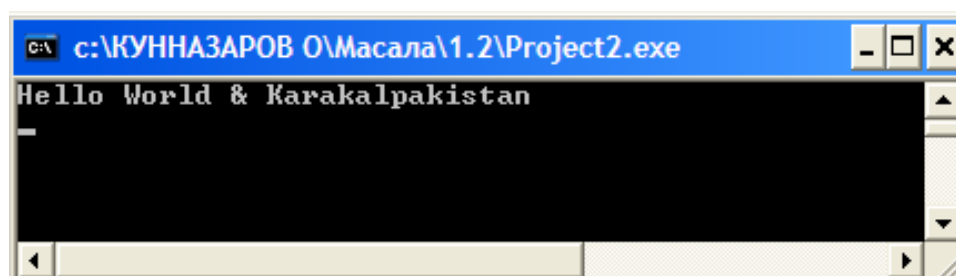
Endi konsol rejimida qo`yidagi loyihani yaratamiz: ekranga – Hello World & Karakalpakistan! so`zin chiqarib bersin. Bu loyihani yaratish algoritmi qo`yidagicha olib boriladi.

Loyiha kodi (**Object Pascal** tilidagi satrlar) kod tahriri oynasida joylashadi. Foydalanuvchi loyiha o`rinlaydigan funktsiyalarin aniqlovchi kodga satr qo`shadi. Loyiha kodin 1.2 papkasida saqlaymiz. 3-rasmda kod tahriri oynasi loyiha kodi bilan ko`rsatilgan.



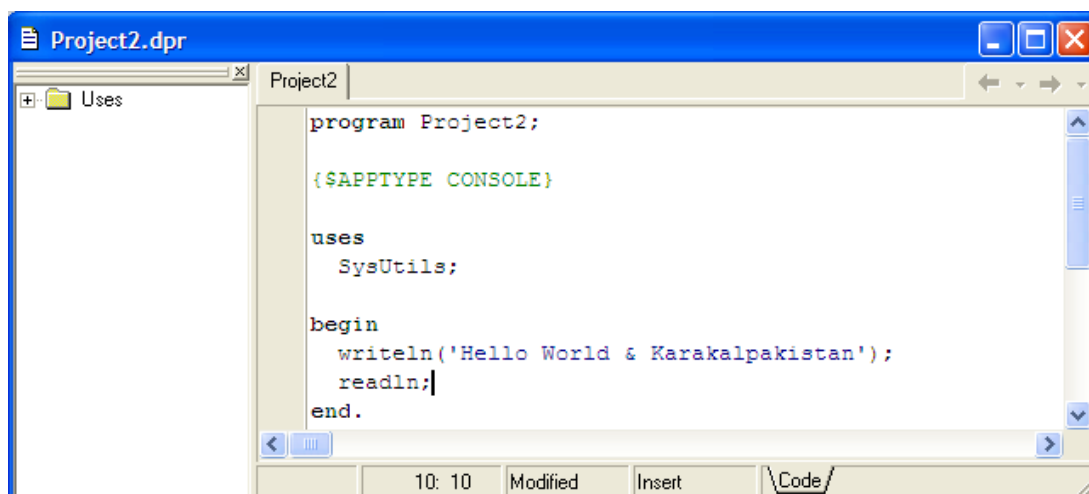
3-rasm.

Loyihani qo`shganimizda 4-rasmda chapda ko`rsatilgan oynani olamiz. Muammo bor–kirillitsa harflarining xato shakllanishi. To`g`ri shakllanishi shu rasmning o`ng tomonida ko`rsatilgan. Xatolik sababi konsolli qo`llanma ICHIM da yaratiladi, bo`erda **ANSI** kodirovkasiga ega **Windows** operatsion tizimi foydalaniladi, u bajarilganda **ASCII** kodirovkasini foydalanuvchi **MS DOS** operatsion tizimiga xabarlik qiladi. Muammoni echish uchun kirillitsa harflarini **ANSI** standartidan **ASCII** standartiga qayta kodirovkalash lozim. Buning uchun Prg1.2 proektiga **ANSI** dan **ASCII** ga satrlarni qayta o`zgartuvchi **Rus** funktsiyasiga ega **RusTrans** moduliga yunalish qo`shamiz.



4-rasm.

Ishlangan o`zgarishlardan so`ng loyiha kodi 5-rasmda ko`rsatilgan. Unda RusTrans moduliga yunalish (murojaat) qo`shilgan, chiqarish operatorida matn satri Rus funksiyasi bilan o`zgaradi.



5-rasm.

1.3. Dastur strukturasi va berilganlar tipi

Jadval-1 da keltirilgan variantlar bilan mos ravishda sonli ma`lumotlar turlarining o`z-aro o`zgaruvchilar dasturin yaratish. Sonli ma`lumotlarni Start turiga Fin1 va Fin2 turlariga haqiqiy va butun sonlar uchun qayta o`zgartish kerak.

Qatorlar va simvoli turlarining o`z-aro o`zgaruvchilar dasturin yaratish. U talaba familiyasini har xil turdagi satrlar bilan shakllantirishi va satrga simvol qo`shishi lozim.

TDateTime turi bilan ishlash dasturin yaratish. U talabaning tug`ilgan vaqtin va sanasi so`rashi, joriy sana/vaqtin chiqarishi, keyingi davom etuvchi o`qishning hoqlagan vaqtin so`rash, o`qishning tamom bo`lish sana/vaqtin va haftadagi kunin chiqarishi lozim.

1-jadval

№	Haqiqiy sonlar				Butun sonlar			
	Start	Fin_1	Fin_2	Fin_3	Start	Fin_1	Fin_2	Fin_3
1	Single	Currency	Extended	Real48	Byte	Cardinal	Integer	Smallint
2	Real48	Single	Currency	Double	Word	Byte	Cardinal	Integer
3	Double	Real48	Single	Real	LongWord	Word	Byte	LongInt
4	Real	Double	Real48	Single	Cardinal	LongWord	Word	Byte
5	Extended	Real	Double	Real48	Shortint	Cardinal	LongWord	Word

6	Comp	Extended	Real	Double	Smallint	Shortint	Cardinal	LongWord
7	Currency	Comp	Extended	Real	Integer	Smallint	Shortint	Cardinal
8	Single	Currency	Comp	Extended	LongInt	Integer	Smallint	Shortint
9	Real48	Single	Currency	Comp	Int64	LongInt	Integer	Smallint
10	Double	Real48	Single	Currency	Word	Int64	LongInt	Integer

Sonli ma'lumotlarni Start turidan Fin1, Fin2 va Fin3 turlariga o'zgartish dasturini yaratish:

1. Haqiqiy sonlar uchun: Start = Real, Fin1 = Extended, Fin2 = Single, Fin3 = Currency.

2. Butun sonlar uchun: Start = Smallint, Fin1 = Cardinal, Fin2 = Longint, Fin3 = Byte. Ma'lumotlar o'lchamlarini nazorat qilish uchun dasturda, argument o'lchamini baytlarga qaytaruvchi **SizeOf** o'rnatilgan funktsiya foydalaniladi. Har bir ma'lumotlar bayt o'lchami va ma'lumot turi ko'rinishida chiqariladi. Sonlar aniqlik uchun belgilangan tochkasiga ega formasi ifodalanadi. Bunda chiqariluvchi o'rinlar (pozitsiyalar) soni beriladi, chiqariluvchi ma'lumotlarni satrda o'rinlarini joylashtirish imkoniyatini beradi.

Dastur listingi

program Prg1.3;

{ \$APPTYPE CONSOLE }

uses

SysUtils, RusTrans;

const

real_0:real=123.456789;

int_0:smallint=-1111;

var

real_1:extended;

real_2:single;

real_3:currency;

int_1:cardinal;

int_2:longint;

int_3:byte;

begin

writeln(' Har xil turdagi haqiqiy sonlar');

writeln;

real_1:=real_0;

real_2:=real_0;

real_3:=real_0;

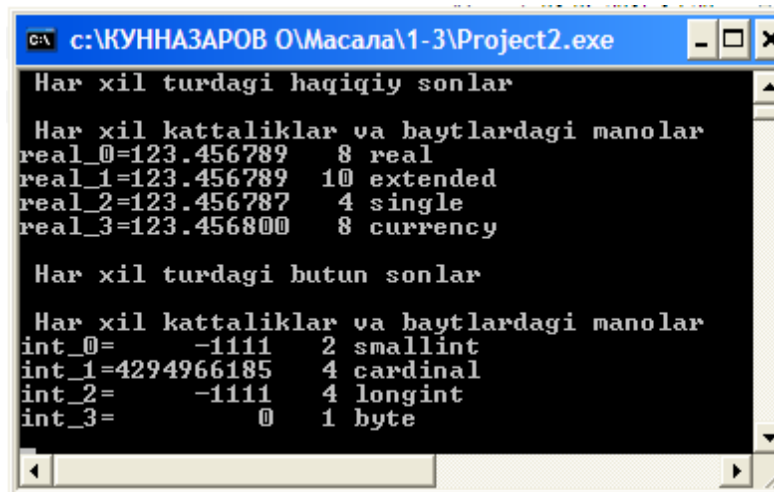
```

writeln(' Har xil kattaliklar va baytlardagi ma'nolar');
writeln('real_0=',real_0:10:6,sizeof(real_0):4,' real');
writeln('real_1=',real_1:10:6,sizeof(real_1):4,' extended');
writeln('real_2=',real_2:10:6,sizeof(real_2):4,' single');
writeln('real_3=',real_3:10:6,sizeof(real_3):4,' currency');
readln;
writeln(' Har xil turdagi butun sonlar');
int_1:=int_0;
int_2:=int_0;
writeln;
writeln(' Har xil kattaliklar va baytlardagi ma'nolar');
writeln('int_0=',int_0:10,sizeof(int_0):4,' smallint');
writeln('int_1=',int_1:10,sizeof(int_1):4,' cardinal');
writeln('int_2=',int_2:10,sizeof(int_2):4,' longint');
writeln('int_3=',int_3:10,sizeof(int_3):4,' byte');
readln;

```

end.

Dastur listingida konstantalar bo`limida **const** turlandirilgan konstantalar deb elon qilingan, boshlang`ich sonlari Start. **var** bo`limida sonning har bir turi uchun o`zgaruvchilar gruppasida Fin1, Fin2, Fin3 elon qilingan. 1-rasmda konsolli rejimning dastur yopilishidan oldingi oyna ko`rinishi berilgan.



1-rasm.

Satrlı va simvolli turdagi o`z-aro o`zgaruvchilar dasturin yaratish. U **World** so`zin har xil turdagi satrlar bilan shakllantirishi va satrga undov belgisi! simvolin qo`shishi lozim.

Dastur listingi

```

program Prg1.4;
{$APPTYPE CONSOLE}

```

```

uses
SysUtils, RusTrans;
var
Str_1:string;
Str_2:string[10];
ShortStr:ShortString;
AnsiStr:AnsiString;
WideStr:WideString;
MyChar:char;
begin
Str_1:='World';
writeln('Har xil turdagi satrlar');
writeln;
Str_2:=Str_1;
ShortStr:=Str_1;
AnsiStr:=Str_1;
WideStr:=Str_1;
writeln(' Har xil kattaliklar va baytlardagi ma'nolar');
writeln;
writeln('AnsiStr =',AnsiStr:7,SizeOf(AnsiStr):5,' AnsiString');
writeln('Str_1 =',Str_1:9,SizeOf(Str_1):5,' String');
writeln('WideStr =',WideStr:7,SizeOf(WideStr):5,' WideString');
writeln('ShortStr =',ShortStr:6,SizeOf(ShortStr):5,' ShortString');
writeln('Str_2 =',Str_2:9,SizeOf(Str_2):5,' String[..]');
readln;
writeln('Satirga simvol kiritish !');
writeln;
MyChar:='!';
Str_2:=Str_2+MyChar;
writeln('Str_2 =',Str_2:9,SizeOf(Str_2):5,' String[..]');
readln;
end.

```

```

c:\KVYHHA3APOB O\Macana\13\Project2.exe
Har xil turdagi satrlar

Har xil kattaliklar va baytlardagi manolar
AnsiStr = World      4 AnsiString
Str_1 =   World      4 String
WideStr = World      4 WideString
ShortStr = World    256 ShortString
Str_2 =   World     11 String[..]

Satirga simvol kiritish !
Str_2 =   World!    11 String[..]

```

2-rasm.

TDateTime turdagi ishchi dasturin yaratish. U tug`ilgan vaqtin va sanasin so`rashi, joriy sana/vaqtin chiqarishi, keyingi davom etuvchi o`qishning xohlagan vaqtin so`rash, o`qishning tamom bo`lish sanasi/vaqtin va haftadagi kunin chiqarishi lozim. 3-rasmda ishning ishlash rejimi oynasi ko`rsatilgan.

Dastur listingi

program Project2;

```
{ $APPTYPE CONSOLE }
```

```
uses
```

```
    SysUtils;
```

```
var
```

```
    Date_1, Date_2, Time, DateTime: string;
```

```
    DateTime_1, DateTime_2: TDateTime;
```

```
    Day: integer;
```

```
begin
```

```
    writeln('Tugulgan kun va yilingiz');
```

```
    readln(Date_1);
```

```
    writeln;
```

```
    writeln('Tugulgan vaqtingiz, soat/daqiqa/soniya');
```

```
    readln(Time);
```

```
    writeln;
```

```
    DateTime := Date_1 + Time;
```

```
    DateTime_1 := StrToDateTime(DateTime);
```

```
    DateTime_2 := Now;
```

```
    writeln(('Bugun'), DateTimeToStr(Now));
```

```
    writeln;
```

```
    writeln('Siz qancha kun oqishni istaysiz?');
```

```
    readln(Day);
```

```
    writeln;
```

```
    DateTime_2 := DateTime_2 + Day;
```

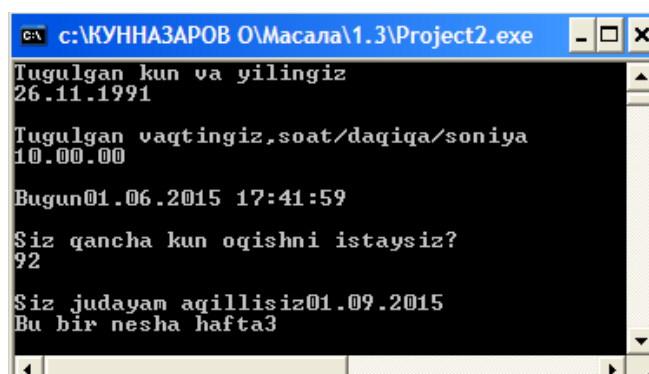
```
    Day := DayOfWeek(DateTime_2);
```

```
    writeln(('Siz judayam aqillisiz'), DateTimeToStr(DateTime_2));
```

```
    writeln(('Bu bir nesha hafta'), Day);
```

```
    readln;
```

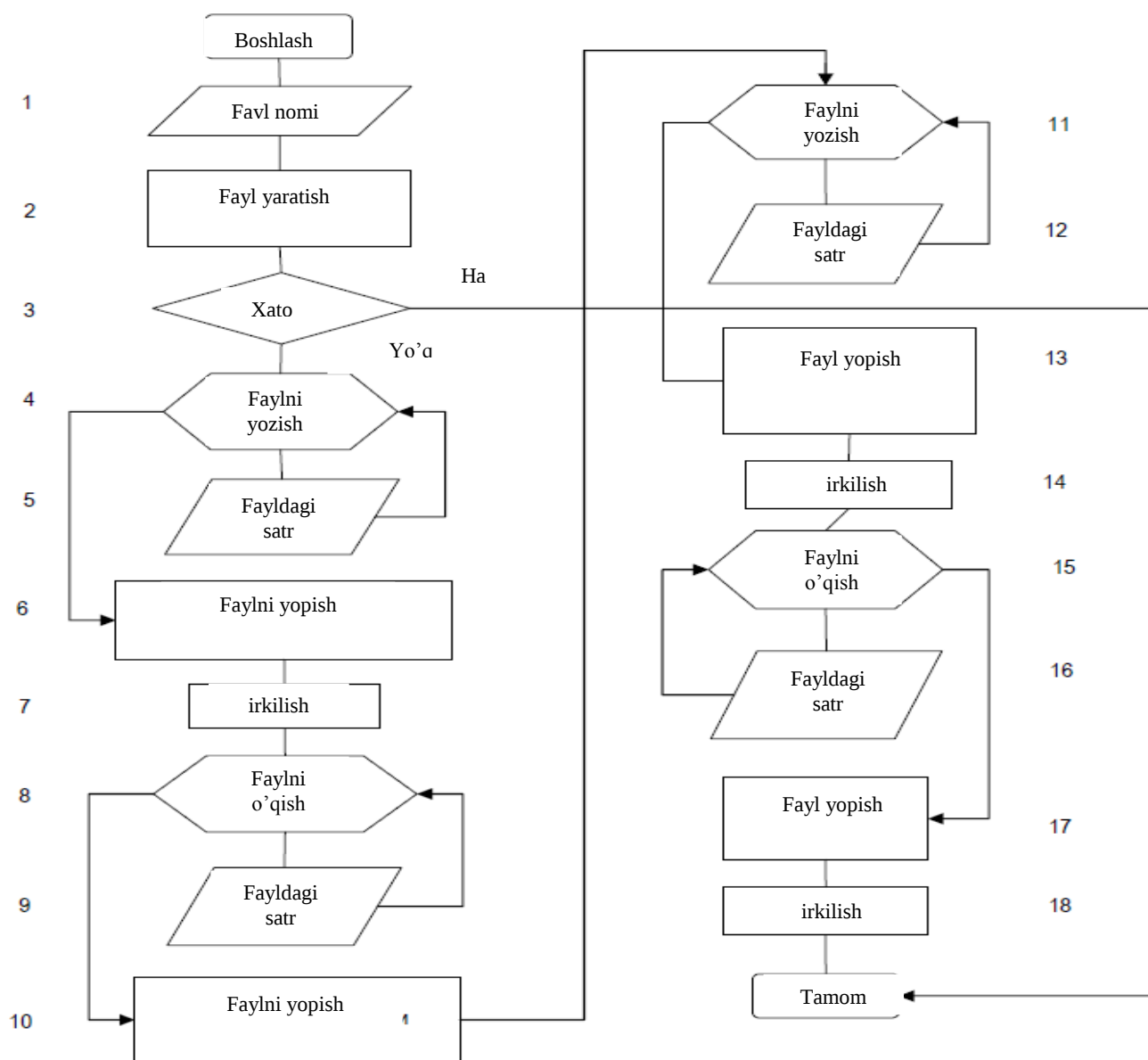
```
end.
```



2-bob. Ma'lumotlar bilan ishlash metodikasi

2.1. Fayllar ustida amallar bajarish

Matnli fayllarni yaratish, yozish va yozishni qo'shish dasturini yaratish. 5 satrdan matnli fayl yaratish va uning ichidagi bor narsasini ko'rish. Keyin shu faylga yana 5 satr qo'shib va yakuniy faylning ichidagi bor narsasini tekshirish. Masalani echish algoritmining blok-sxemasini 1-rasmda keltirilgan.



1-rasm.

1-blokda diskdagi fayl atamasi kiritiladi. 2-blokda papkaning belgisi atamasidan (dastur kodida FileDir konstantasi) va shu atamada faylga yo'l ishlab chiqiladi, unnan so'ng fayl yaratiladi. Agar xatolik kelip chiqsa, unda 3-blokda dastur yakunlanadi.

Konsolli qo'llanmada fayllar bilan ishlaganda fayllarga yo'l ishlab chiqarilganda kirillitsa simvollarin foydalanish mumkin emas.

Agar xatolik yuz bermasa, unda 4 blokda **Imax** fayliga satrlar kiritish buyicha tsikl ochiladi. Satrlarni kiritish 5 blokida amalga oshiriladi. Ma'noga ega satrlarni kiritish tavsiya etiladi, eng yaxshisi sonli - 1, 2, 3...Imax. 6 blokda fayl yopadi, keyinchalik tsiklning 8 blokida tashkillashtiriladigan 9 blok orqali **Imax** satrlarin satrma satr chiqarish yo'li bilan tekshirish uchun ochiladi. 10 blokda fayl yana yopiladi, so'ngra tsiklning 11 blokida amalga oshiriladigan 12 blokining yana **Imax** satrlarin fayl so'ngina satrma satr kiritish yo'li bilan to'liqtirilishi uchun ochiladi. 13 blokida fayl yopiladi, so'ngra tsiklning 15 blokida tashkillashtiriladigan 16 blok tufayli barcha **Imax*2** satrlarin satrma satr chiqarish yo'li bilan tekshirish uchun ochiladi. Jarayonni kuzatish uchun algoritmgaga pauzalar kiritilgan (7, 14 va 18 bloklari). 2-rasmda ko'rsatilgan.

Dastur listingi

```
program Prg 2.2;
{$APPTYPE CONSOLE}
uses
  SysUtils, RusTrans;
const
  FileDir='c:\users\PO_11\Ivanov\08_1\';
var
  MyFile: text;
  FilePath, FileName, Temp: string;
  i, imax: integer;
label
  Exit;
begin
  writeln('Fayl nomini kiriting');
  readln(FileName);
  writeln;
  FilePath:=FileDir+FileName;
  AssignFile(MyFile,FilePath);
  {$I-}
  Rewrite(MyFile);
  {$I+};
  if IOResult = 0
  then writeln('Raxmat')
  else
```

```

begin
writeln('Xato');
readln;
goto Exit
end;
writeln;
imax:=5;
writeln('Kiriting '),imax,(' Satir');
for i:=1 to imax do
begin
write(' Satir ',i,' ');
readln(Temp);
writeln(MyFile,Temp);
end;
CloseFile(MyFile);
Reset(MyFile);
i:=1;
writeln;
writeln('Berilgan fayl');
writeln;
while not EOF(MyFile) do
begin
readln(MyFile,Temp);
writeln(' Satir ',i,' ',Temp);
i:=i+1
end;
writeln;
CloseFile(MyFile);
{$I-}
Append(MyFile);
{$I+};
if IOResult = 0
then writeln('OK')
else
begin
writeln('Xato');
readln;
goto Exit
end;
writeln;
writeln('Yana kiriting ',imax,' Satir');
for i:=imax+1 to imax*2 do
begin
write(' Satir ',i,' ');

```

```

readln(Temp);
Writeln(MyFile,Temp)
end;
CloseFile(MyFile);
Reset(MyFile);
i:=1;
writeln;
writeln('Berilgan fayl');
writeln;
while not EOF(MyFile) do
begin
readln(MyFile,Temp);
writeln(' Satir ',i,' = ',Temp);
i:=i+1
end;
writeln;
writeln('OK');
readln;
CloseFile(MyFile);
Exit;;
end.

```

```

c:\KYHHA3APOB O\Macana\2.2\Project2.exe
Fayl nomini kiriting
InFile
Raxmat
Kiriting 5 Satir
Satir 1 = 1
Satir 2 = 2
Satir 3 = 3
Satir 4 = 4
Satir 5 = 5
Berilgan fayl
Satir 1 = 1
Satir 2 = 2
Satir 3 = 3
Satir 4 = 4
Satir 5 = 5
OK
Yana kiriting 5 Satir
Satir 6 = 6
Satir 7 = 7
Satir 8 = 8
Satir 9 = 9
Satir 10 = 10
Berilgan fayl
Satir 1 = 1
Satir 2 = 2
Satir 3 = 3
Satir 4 = 4
Satir 5 = 5
Satir 6 = 6
Satir 7 = 7
Satir 8 = 8
Satir 9 = 9
Satir 10 = 10
OK

```

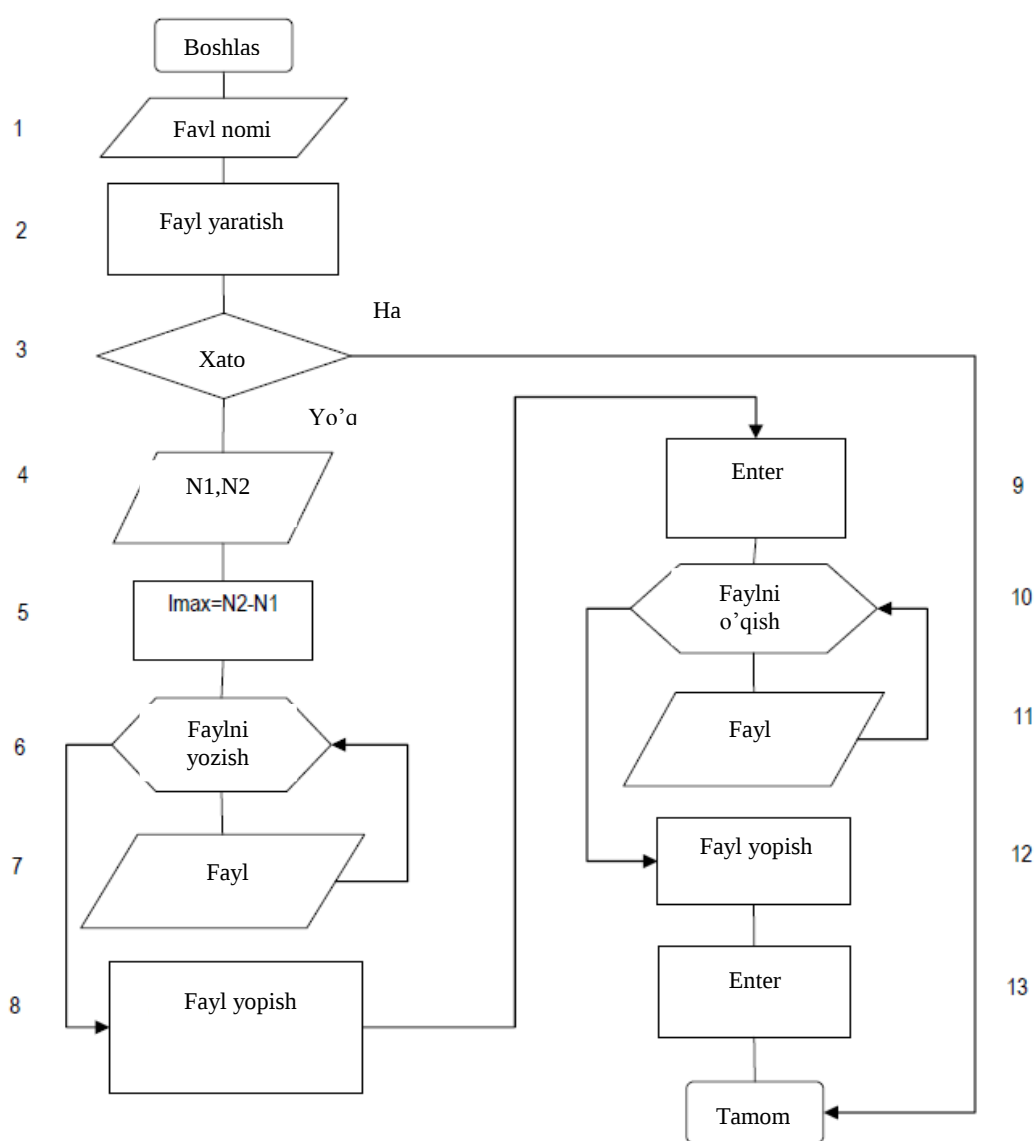
2-rasm.

Fayl elementlariga erkin kirishni ko'rsatuvchi dastur yaratish. U faylga ketma-ket N1 dan N2 gacha diapazonida (butun va o'ng) sonlarni kiritadi. Yozib

bo'lgannan so'ng fayldan barcha nomerlari bilan (0 dan boshlab) komponentlari displayda o'qiladi.

Masalani echish algoritmining blok-sxemasi 3-rasmida keltirilgan.

1blokda foydalanuvchi savoli buyicha diskga fayl atamasin kiritadi (masalan,e). 2blokda fayl yozilishi lozim bo'lgan (dastur kodidagi **FileDir** konstantasi) nomi belgili papkasidan, va bul atamadan faylga yo'l ishlab shiqiladi, so'ngra fayl yaratiladi. Agar xatolikka yo'l qo'yilgan bo'lsa, unda 3 blokda dastur **Error** xabari bilan tamomlanadi – Xatolik. Ishlab turgan qo'llanma oynasi 4-rasmida ko'rsatilgan.



3-rasm.

Teskari paytda 4 blokda foydalanuvchi ikki butun o`ng N1 va N2 sonlarin kiritadi. 5 blokda  $I_{max} = N2 - N1$  faylga kiritishi kerak bo`lgan komponentlarining soni aniqlanadi. 6 blok **I_{max}** komponentlarin yozish tsiklin ochadi, uning tanasida 7 blok N1..N2 intervalida faylga sonlarni yozishni amalga oshiradi. 8 blokda fayl yopiladi, keyin tsiklning 10 blokida tashkillashtiriladi 11 blok tufayli I_{max} komponentlarin fayldan nomerlari bilan birga satrma satr chiqarish yuli bilan tekshirish uchun ochiladi. 12 blokda fayl yopiladi. Jarayonni kuzatish uchun algoritmgga pauzalar kiritilgan (9 va 13 bloklari).

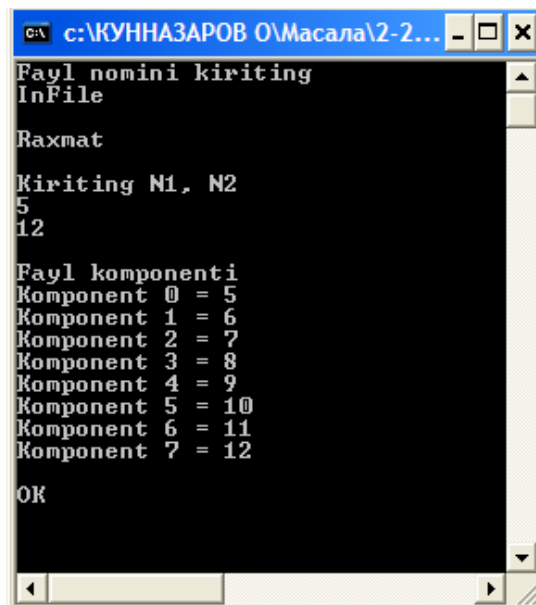
Dastur listingi:

```
program Prg2.4;
{$APPTYPE CONSOLE}
uses
  SysUtils, RusTrans;
label
  Exit;
const
  FileDir='c:\users\PO_11\Ivanov\08_2\';;
var
  MyFile: File of integer;
  FilePath, FileName: string;
  N1, N2, i, imax, Temp: integer;
begin
  writeln('Fayl nomini kiriting');
  readln(FileName);
  writeln;
  FilePath:=FileDir+FileName;
  AssignFile(MyFile,FilePath);
  {$I-}
  Rewrite(MyFile);
  {$I+};
  if IOResult = 0
  then writeln('Raxmat')
  else
  begin
    writeln('Xato');
    readln;
    goto Exit
  end;
  writeln;
  writeln('Kiriting N1, N2');
```

```

readln(N1,N2);
writeln;
imax:=N2-N1;
for i:=0 to imax do
begin
Temp:=N1+i;
write(MyFile,Temp)
end;
CloseFile(MyFile);
Reset(MyFile);
writeln('Fayl komponenti');
for i:=0 to imax do
begin
Seek(MyFile,i);
read(MyFile,Temp);
writeln('Komponent ',i,' = ',Temp);
end;
CloseFile(MyFile);
writeln;
writeln('OK');
readln;
Exit;;
end.

```



4-rasm.

Asosiy faylni fayl qabul qiluvchiga yuqori tezlikda ko`chirish uchun dastur yaratish. Asosiy fayl sifatida oldin yaratilgan Project2.exe dasturi kompilyatsiyasi natijasin foydalanish. Fayl qabul qiluvchi sifatida Project2.exe atamali faylin belgilash. Bajirilgandan sung diskta shu o`lchamdagi va ma`nodagi fayl qabul

qiluvchi bor ekanin tekshirish lozim. Bunga ishonch qabul qilish uchun shu faylni bajarish mumkin.

Dastur listingi:

```
program Prg2.3;
{$APPTYPE CONSOLE}
uses
  SysUtils,RusTrans;
label
  Exit;
const
  FileDir=' c:\KUNNAZAROV O\Masala\2.2\ ';
  FileDir=' c:\KUNNAZAROV O\Masala\2.3\';
var
  InFile,OutFile: File;
  Buf: array[1..512] of byte;
  FilePath1,FilePath2: string;
  i,j: integer;
begin
  FilePath1:=FileDir1+'prg2.3.exe';
  AssignFile(InFile,FilePath1);
  FilePath2:=FileDir2+'prg2.3a.exe';
  AssignFile(OutFile,FilePath2);
  Reset(InFile,512);
  Rewrite(OutFile,512);
  writeln('Fayl istochnik ',FilePath1);
  writeln('Fayl priemnik ',FilePath2);
  writeln;
  writeln('Faylni nusxalash?');
  readln;
  writeln(' Nusxalashni boshlash');
  writeln;
  while not EOF(InFile) do
  begin
    BlockRead(InFile,Buf,1,i);
    BlockWrite(OutFile,Buf,1,j)
  end;
  writeln(' Nusxalash yakunlandi');
  readln;
```

```
CloseFile(InFile);  
CloseFile(OutFile);  
end.
```

Fayl bilan ma'lumot almashish dasturin yaratish. U buferge 6 tasodifiy sonlarni kiritadi va ularni faylga ko'chiradi (u loyiha papkasida yaratiladi). Keyin bufer nollarni, va fayldan birinchi 4 sonlari qaytadi. Ishlab turgan qo'llanmaning oynasi 5-rasmda ko'rsatilgan ko'rsatilgan.

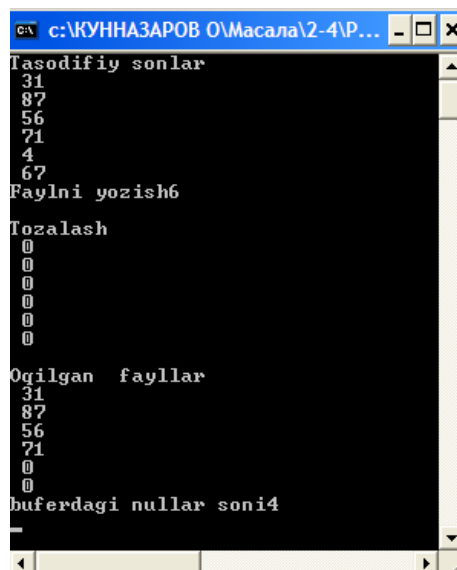
Dastur listingi:

```
program Prg2.4;  
{ $APPTYPE CONSOLE }  
uses  
SysUtils, RusTrans;  
const  
Count=6;  
var  
hMyFile: Integer;  
Buffer: Array[1..Count] of Byte;  
i, Fact: Integer;  
begin  
hMyFile:=FileCreate('MyFile');  
writeln('Tasodifiy sonlar');  
Randomize;  
for i:=1 to Count do  
  begin  
    Buffer[i]:= Random(100);  
    writeln(' ', Buffer[i]);  
  end;  
Fact:=FileWrite(hMyFile, Buffer, Count);  
writeln('Faylni yozish',Fact);  
readln;  
writeln('Tozalash');  
for i:=1 to Count do  
  begin  
    Buffer[i]:=0;  
    writeln(' ',Buffer[i]);  
  end;  
readln;
```

```

FileSeek(hMyFile, 0, 0);
Fact:=FileRead(hMyFile, Buffer, Count-2);
writeln('O`qilgan fayllar');
for i:=1 to Count do
begin
writeln(' ',Buffer[i]);
end;
writeln(' buferdagi nullar soni '),Fact);
FileClose (hMyFile);
readln;
end.

```



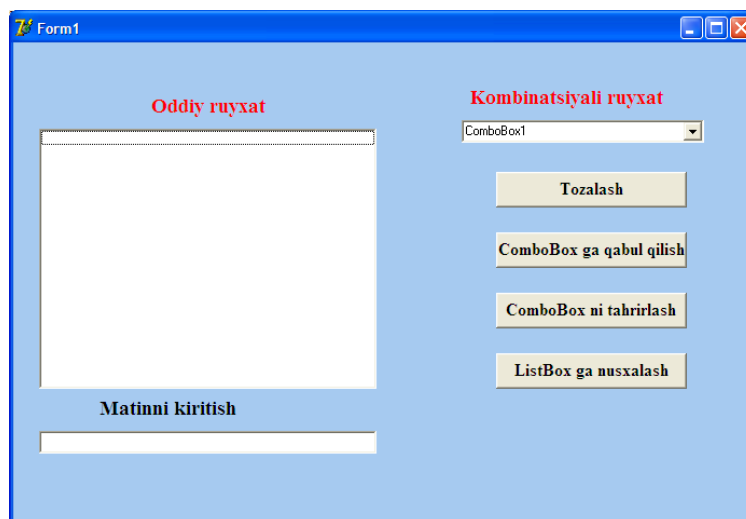
5-rasm.

Endi loyiha rejimida ishlashni ko`rib chiqamiz.

2.2. Delphi da ro`xatlar

Ro`yxatlar bilan ishlash proektin yaratish. Loyihada nazarda tutish:

1. Kiritish maydonidan kombinatsiyalangan ro`yxatni to`ldirish.
2. Kombinatsiyalangan ro`yxatni tahrirlash (satr kiritish, satrni ko`chirish).
3. Satrlarni **ComboBox** ro`yxatidan **ListBox** ro`yxatiga ko`chirish. Dasturning tavsiya etilgan Formasi 1-rasmda keltirilgen.



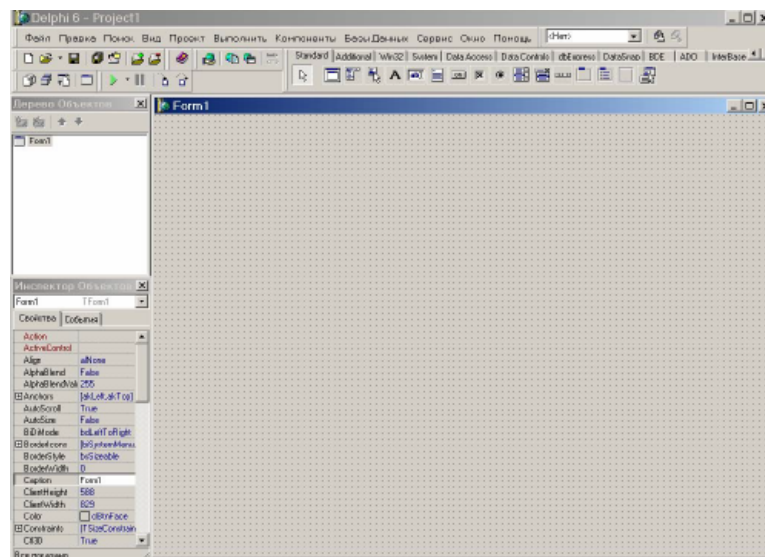
1-rasm

Forma qo`yidagi komponentlarni o`z ichiga oladi:

- **ListBox1** – oddiy ro`yxat.
- **Label1** – **ListBox1** komponentining boshlamasi uchun belgi.
- **ComboBox1** – kombinatsiyalangan ro`yxat.
- **Label2** – **ComboBox1** komponentining boshlamasi uchun belgi.
- **Edit1**- matnni kiritish uchun bir satrli tahrir. Unga ro`yxat uchun satr matni kiritiladi.
- **Label3** – **Edit1** tahririning boshlamasi uchun belgi.
- **Button1**– "Tozalash" tugmasi. **Edit1** maydonin tozalash va u erga kursorni o`rnatish imkoniyatin beradi.
- **Button2** – **ComboBox** da "Qabul qilish" tugmasi **ComboBox1** komponentin to`ldirish uchun.
- **Button3** – "Tahrirlash **ComboBox**" tugmasi. Ro`yxatda satrlarni kiritish, olib tashlash, o`z-aro almashtirish va ko`chirish imkoniyatin beradi.
- **Button4** – "Nusxalash **ListBox**" tugmasi. **ComboBox** da tanlanadigan satrlarni **ListBox** qa ko`chirish imkoniyatin beradi.

Windows provodniginda **Delphi-7** ni tanlab aktivlashtiramiz.

Ekran 2-rasmda berilganiday ko`rinishga ega. U komponentlar bilan to`ldirilishi kerak bo`lgan bo`sh formasi chiqariladi. Ob`ektlar daraxti oynasida forma komponenti shakllangan **Form1** atamasida.



2-rasm.

Forma xossalarin berish algoritmi qo'yidagicha:

Form1 ob'ektin tanlaymiz va uning o'lchamlarin va Formasin yaratamiz Ob'ektlar inspektori oynasida Forma xossalarin belgilaymiz:

Form1=> Prg2.7. Formada boshlama o'zgaradi

Formani to'ldirish. Formani to'ldirishda komponentalar palitrasida kerakli blok tanglanadi va Formaga ko'chiriladi. U erda uning o'rni va o'lchamlari yig'iladi. Keyin ob'ektlar inspektori oynasida komponent xossalari belgilanadi. Xossalarning ko'pchilik bo'lagi avtomat turda o'rnatiladi, ularni tahrirlash kerak emas. Bizga kerak komponentalar **Standard** komponentalar palitrasida joylashadi. Knopkalar uchun, hodisalarin qayta ishlashi kodin yozib funktsionallik qo'shish zarur. Knopkaga ikki marta bosishda Delphi-7 avtomat turda qayta ishlashining ilova dasturin chiqaradi. Loyihaning formasi moduli kodina kiritiladi:

1. Ob'ekt maydonina qayta ishlashshining boshlamasi. Qayta ishlashining boshlamasi avtomat turda shakllanadi. Uning atamasi qayta ishlashi prefiksi on bilan komponenta va hodisa atamasin probellarsiz yozilganlarin o'z ichiga oladi.
2. Amalga oshirish bo'limida qayta ishlash shabloni. U boshlama boshinda Forma atamasin qo'shish bilan ob'ekt maydonindagi boshlama bilan mos keluvchi boshlamaga ega. Unga **begin...end** operator ichiga kichik dasturining matnin kiritish kerak. Bazi bir qayta ishlashlar uchun lokal o'zgaruvchilar talab etilishi mumkin.

Dastur listingi:**unit** Prg2.8;**interface****uses**

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

typeTPrg2.8 = **class**(TForm)

ListBox1: TListBox;

ComboBox1: TComboBox;

Button1: TButton;

Edit1: TEdit;

Button2: TButton;

Label1: TLabel;

Label2: TLabel;

Label3: TLabel;

Button3: TButton;

Button4: TButton;

procedure Button1Click(Sender: TObject);**procedure** Button2Click(Sender: TObject);**procedure** Button3Click(Sender: TObject);**procedure** Button4Click(Sender: TObject);**private**

{ Private declarations }

public

{ Public declarations }

end;**var**

Prg2.8: TPrg2.8;

implementation

{\$R *.dfm}

procedure TPrg2.8.Button1Click(Sender: TObject);**begin**

Edit1.Text:='';

Edit1.SetFocus;

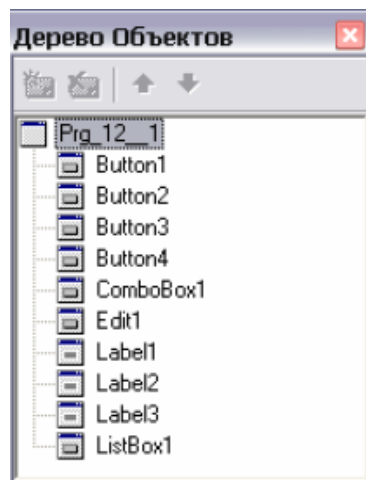
end;**procedure** TPrg2.8.Button2Click(Sender: TObject);**begin**

```

ComboBox1.Items.Add(Edit1.Text);
Edit1.Text:="";
Edit1.SetFocus;
end;
procedure TPrg2.8.Button3Click(Sender: TObject);
begin
ComboBox1.Items.Insert(comboBox1.ItemIndex,Edit1.Text);
Edit1.Text:="";
Edit1.SetFocus;
end;
procedure TPrg2.8.Button4Click(Sender: TObject);
begin
ListBox1.Items.Add(ComboBox1.Items[ComboBox1.ItemIndex]);
end;
end.

```

Panellarni komponentalar bilan tultirgannan so`ng forma 2.8-rasmdagi ko`rinishga ega bo`ladi. Ob`ektlar daraxti oynasi 3-rasmdagi ko`rinishga ega bo`ladi. U buyicha kiritilgan komponentalar tuzilg`in, ularning atamalarin va ierarxiyasini tekshiring.



3-rasm

4. Loyihani 2.8 atamasiga ega papkada saqlaymiz.

Buning uchun dialog oynasining hosil bo`lishiga olib keluvchi Fayl => Soxranit` Proekt Kak bo`yrug`in ishlatamiz. Unnan kerakli papkag`a tushib loyihani saqlash

kerak. Qo'llanmaning proektin saqlaganda loyihani (Prg2.8 atamasi bilan) va Forma modulini (Prg2.8 atamasi bilan) saqlash zarur.

5. Dasturning ishlashini tekshirib ko'ramiz. Dasturni «Bajarish» bo'yruqi bilan ishga qo'shamiz. Rejim oynasi hosil bo'ladi.

- Initsializatsiyalash. Initsializatsiya tugmasi bilan **Edit1** yangilaymiz va og'an kiritish foksini beramiz. Kursor **Edit1** kiritish maydonida hosil bo'ladi.
- **ComboBox** ro'yxatini tuzish. **Edit1** tahririda har bir satr uchun matn kiritamiz, Prinyat tugmasi bilan **ComboBox** da uni ro'yxatga ko'chiramiz. Operatsiyani kerakli son marta takrorlaymiz. Har bir marta **ComboBox** ro'yxatini tekseremiz, shunda yangi satr hosil bo'lganiga ishonich hosil etib.
- Tahrirlash. **Edit1** tahririda erkin tanglanuvchi pozitsiyasiga kiritish uchun matn satrini kiritamiz. **ComboBox** da kiritish uchun pozitsiya ajratamiz, uning matni tahrirlashning yuqori satrini almashtiradi.
- Ko'chirish. **ComboBox** da ko'chirish uchun pozitsiya ajratamiz, uning matni tahrirlashning yuqori satrini joy almashtiradi. **ListBox** da Kopirovat tugmasi bilan **ListBox** ga satrni ko'chiramiz. Har bir ret **ListBox** ni, u erda navbattagi pozitsiyasida yangi satr hosil bo'lganiga ishonch hosil etish uchun tekshiramiz.

2.3. Jadvallar ustida amallar

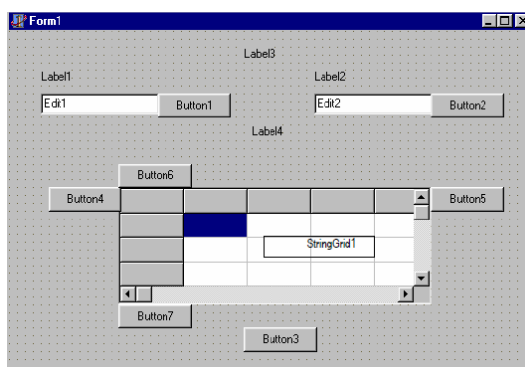
Endi "Jadval konvertatsiyasi" proektini yaratishni ko'rib chiqamiz, loyiha rejimi asosida. Jadvalda yechayka ichidagisini tahrirlash, satr va ustunlarni qo'shish va olib tashlash mumkinchiliklarini ko'zda tutish kerak. Topshiriqlar varianti 2-jadvadda keltirilgan.

2-Jadval

№	Valyuta		
1	AQSH dollari	Funt sterlingov	Evro
2	Funt sterlingov	Evro	So'm
3	Evro	Yaponskaya yen	AQSH dollari

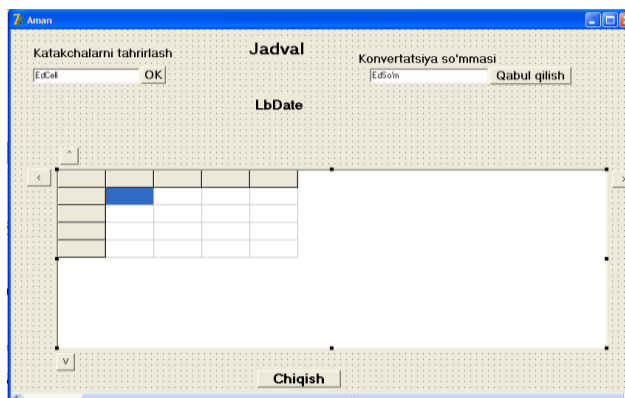
Topshiriqda davlatlar aro ishlatiladigan valyutalarni konvertatsiya etishni ko`rib chiqamiz. Jadvaldan foydalanib Uzbekiston so`min AQSh, Byuuk Britaniya va Evropa birlashmasi valyutalariga konvertatsiyasi uchun “Jadval konvertatsiyasi” proektin yaratish talab etiladi. Buning uchun qo`yidagi algoritmni bajaramiz.

1. **Delphi** ni yuklash.
2. Unda yangi loyiha rejimiga o`tish.
3. Form1 komponentining **Caption** xossalariga Pereschet denejnix summ ma`nosin biriktirish.
4. Modulni Prg_14_1_, atamasida, loyihani Prg_14_1 atamasida saqlash.
5. Forma komponentalarin o`rnatish: **Edit1** va **Edit2** bir satrli tahrirlar, belgilari **Label1** - **Label4**, **Button1**- **Button7** knopkilari (Standard betidan), **StringGrid1** satrlar jadvali (Additional betidan). Komponentlarni 1-rasmda ko`rsatilganiday joylashtirish.
6. Komponentlarning keyingi xossalarin berish:



1-rasm

Yuqorida keltirilgan o`rnatishlardan so`ng loyiha interfeysi 2-rasmdagi ko`rinishga ega bo`ladi.



2-rasm

Loyiha Formasi listingi:

unit Prg_14_1_;

interface

uses

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

type

TForm1 = **class**(TForm)

StringGrid1: TStringGrid;

EdCell: TEdit;

EdSum: TEdit;

SellAgree: TButton;

SummAgree: TButton;

Exit: TButton;

DecCol: TButton;

IncCol: TButton;

DecRow: TButton;

IncRow: TButton;

Label1: TLabel;

Label2: TLabel;

Label3: TLabel;

Label4: TLabel;

procedure FormCreate(Sender: TObject);

procedure DecColClick(Sender: TObject);

procedure IncColClick(Sender: TObject);

procedure DecRowClick(Sender: TObject);

procedure IncRowClick(Sender: TObject);

procedure ExitClick(Sender: TObject);

procedure SellAgreeClick(Sender: TObject);

procedure StringGrid1DblClick(Sender: TObject);

procedure SummAgreeClick(Sender: TObject);

private

{ Private declarations }

public

{ Public declarations }

end;

var

Form1: TForm1;

implementation

{\$R *.DFM}

procedure TForm1.FormCreate(Sender: TObject);

begin

StringGrid1.ColCount:=6;

```

StringGrid1.RowCount:=4;
StringGrid1.Cells[0,0]:='Valyuta';
StringGrid1.Cells[1,0]:='Kurs so'm';
StringGrid1.DefaultColWidth:=78;
StringGrid1.Cells[2,0]:='Sumdagi chiqishi';
StringGrid1.Cells[3,0]:='Komissiya, %';
StringGrid1.Cells[4,0]:='Komissiya';
StringGrid1.Cells[5,0]:='To'lam,So'm.';
StringGrid1.Cells[0,1]:='USA';
StringGrid1.Cells[0,2]:='FST';
StringGrid1.Cells[0,3]:='EUR';
StringGrid1.cells[1,1]:='31,5';
StringGrid1.cells[1,2]:='45';
StringGrid1.cells[1,3]:='32,1';
StringGrid1.cells[3,1]:='2';
StringGrid1.cells[3,2]:='2';
StringGrid1.cells[3,3]:='2';
Label4.Caption:='Bugungi holat '+DateToStr(Date);
end;
procedure TForm1.DecColClick(Sender: TObject);
begin
if StringGrid1.ColCount>2 then StringGrid1.ColCount:=StringGrid1.ColCount-1;
end;
procedure TForm1.IncColClick(Sender: TObject);
begin
StringGrid1.ColCount:=StringGrid1.ColCount+1;
end;
procedure TForm1.DecRowClick(Sender: TObject);
begin
if StringGrid1.RowCount>2 then
StringGrid1.RowCount:=StringGrid1.RowCount-1;
end;
procedure TForm1.IncRowClick(Sender: TObject);
begin
StringGrid1.RowCount:=StringGrid1.RowCount+1;
end;
procedure TForm1.ExitClick(Sender: TObject);
begin
Close;
end;
procedure TForm1.SellAgreeClick(Sender: TObject);
begin
StringGrid1.Cells[StringGrid1.Col,StringGrid1.Row]:=EdCell.Text;
end;

```

```

procedure TForm1.StringGrid1DblClick(Sender: TObject);
begin
  if (StringGrid1.Col=1) and (StringGrid1.Row>0)
  then EdCell.Text:=StringGrid1.Cells[StringGrid1.Col,StringGrid1.Row];
end;
procedure TForm1.SummAgreeClick(Sender: TObject);
var
  ResVal, Komis:real;
  i:byte;
  S:string;
begin
  with form1 do
    for i:=1 to StringGrid1.RowCount-1 do
      if (StrToFloat(StringGrid1.Cells[1,i])<>0) then
        begin
          Komis:=StrToFloat(EdSum.Text)/100*
            StrToFloat(StringGrid1.Cells[3,i]);
          ResVal:=(StrToFloat(EdSum.Text)-Komis)/
            StrToFloat(StringGrid1.Cells[1,i]);
          str(Komis:0:2,s);
          StringGrid1.Cells[4,i]:=s;
          str(ResVal:0:2,s);
          StringGrid1.Cells[2,i]:=s;
          str(StrToFloat(EdSum.Text):0:2,s);
          StringGrid1.Cells[5,i]:=s;
        end;
      end;
    end.

```

Dasturda qo`yidagi usullar foydalanilgan:

1. FormCreate. U formani yaratishda ACol (ustun nomeri) va ARow (satr nomeri) yacheykalari koordinatalari buyicha jadvalga boshlang`ish ma`lumotlarni kiritadi. **StringGrid1** komponentasining bo`lak yacheykalariga **StringGrid.Cell [ACol, ARow]** xossasi orqali matnli ma`nolari, agar **Label4** belgisining **Caption** xossasiga joriy vaqt ma`nosi biriktiriladi.
2. CellAgreeClick. U **EdCell** matnli "Yacheykalar tahriri" dan ma`lumotlarni **SellAgree** (OK) tugmasi boshqaruvida jadvalning ajratilgan yacheykalarining joriy koordinatalari buyicha jadvalga kiritadi. Xisobga oling, ma`lumotlar matnli informatsiya sifatida kiritiladi.

3. **StringGrid1DblClick**. Jadval yacheykalariga sichqonchani ikki marta boshish orqali ma'lumotlarni jadvaldan "Yacheykalar tahriri" **EdCell** oynasiga, ularning keyinchalik tez tahrirlanishi uchun ko'chirish mumkinchiligin beradi. Masalan, komissiya foyizi yoki kursning o'zgarishidagi. Joriy yacheyka koordinatalariga kirish **StringGrid1** komponentining **Col** va **Row** xossalari yordamida ta'miynlanadi. Faqatgina valyuta kurslarin tahrirlash kerak.

4. **SumAgreeClick**. Uning yordamida " konvertatsiya so'mmasi" **EdSum** matnli tahriridan ma'lumotlar **SumAgree** tugmasini boshkanida jadvalga ag'imdag'i koordinatalri buyicha kiritiladi. Xisobga oling, ma'lumotlar tahrir tarafidan matnli informatsiya sifatida qabul qilinadi, faqat xisoblashlarni sonlar bilan amalga oshirish zarur.

5. Jadval o'lchamlarin o'zgartish uchun **ColCount** va **RowCount** xossalari foydalaniladi. Keyingi 4 usuli jadval o'lchamlari tugmalariga bosish orqali (**onClick** hodisasin chaqiradi) dasturning bajarilish protsessida ustunlar va satrlar ko'payadi yoki kamayadi:

- **TForm1.DecColClick** ustinlar sonin kamaytirish.
- **TForm1.IncColClick** ustinlar sonin ko'paytirish.
- **TForm1.DecRowClick** satrlar sonin kamaytirish.
- **TForm1.IncRowClick** satrlar sonin ko'paytirish.

6. Dastur bilan ishlashni yakunlash uchun **ExitClick** usulin foydalanishni amalga oshirsa asosiy forma yopiladi. **Windows** oynasin yopish tugmasi takrorlanadi.

Loyihani bajarishni ishga qo'shish va barcha tugmalarning harakatin tekshirish. **EdSum** ga konvertatsiya uchun so'mma kiritish va mos yacheykalarda xisoblash natijalarin nazoratga olish. Valyuta kursin o'zgartish va buning hisoblashlar natijalariga qanday tasir etishin nazoratga olish. Bajarishda forma 3-rasmdagi ko'rinishga ega bo'ladi.

Aman

Katakchalarni tahrirlash

OK

Jadval

Konvertatsiya so'mmasi

Qabul qilish

Bugungi holat 01.06.2015

<	Valyuta	Sondagi kursi	Sondagi chiqish	Komissiya, %	Komissiya	Tolam, Som.	>
	USA	2870		2			
	FST	3240		2			
	EUR	2970		2			

v

Chiqish

4-rasm

3-bob. Amaliy masalalarga tajriba mashg'ulotlarin o'tkazish metodikasi

3.1. Funktsiyalar grafigin chizish

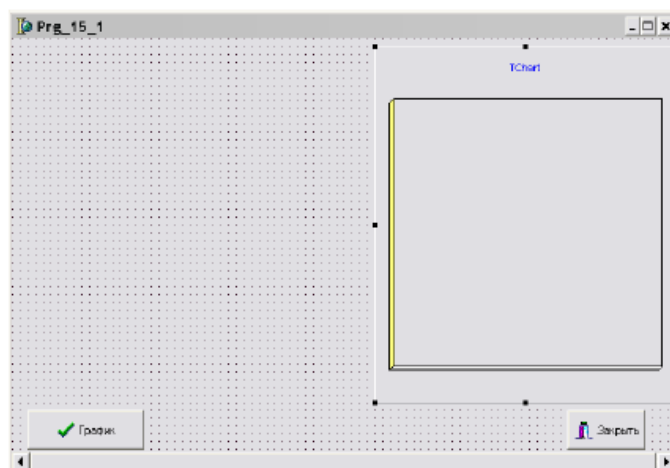
Funktsiyalar grafiglari formasida rasm solish va matnin chiqarish dasturin yaratish. Masala 1-jadvalda keltirilgen.

1-Jadval.

№	Funktsiya grafigi $y(x)$		
	Funktsiya	x boshlanishi	x tamomlanishi
1			
2	$\sin(x)$	0	6π
3	$\cos(x)$	0	6π
4	$\sin(x) + \sin(2x)$	0	6π
5	$\sin(x) - \sin(2x)$	0	6π
6	$\sin(x) + \cos(x)$	0	6π
7	$\sin(x) + \cos(2x)$	0	6π
8	$\exp(x)$	0	2
9	$\exp(-x)$	0	2
10	$\ln(x)$	1	10

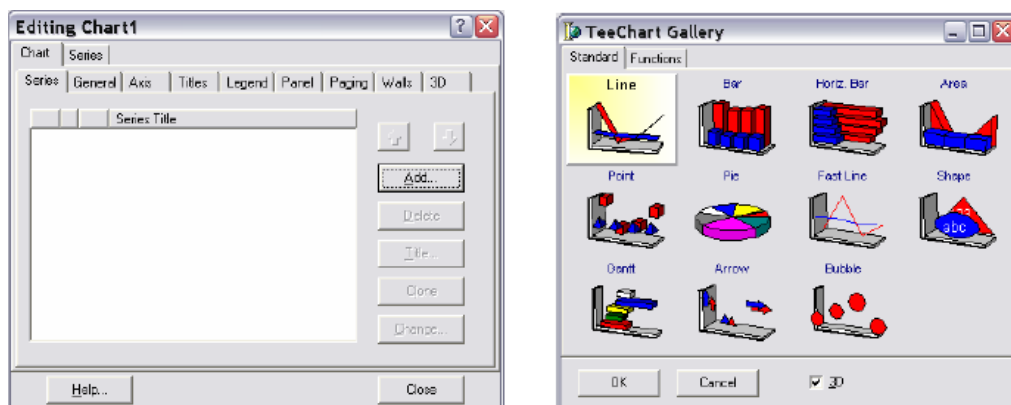
Funktsiya grafigi formasida tushuntiruvchi matniga ega rasm chizish dasturin yaratish. Funktsiya -sinus. Dastur grafigin chizishning uch variantin bor:

- Nuqta bilan **Pixels** (grafik oynaning chap tepa qismida) xossasi yordamida.
- To'g'ri chiziqlar bilan **LineTo** (grafik oynaning pastki chap bo'limida) usulin foydalanish orqali.
- **TChart** (oynaning o'ng bo'limida) komponentasin foydalanish orqali.
- 3.1-rasmda forma oynasi konstruktorlanish boshlanishinda ko'rsatilgan. Formada keyingi komponentalar joylashgan:
- **BitBtn1** (Grafika) tugmasi **Additional** betidan. U grafiglarning oynada tuzilishin chaqiradi.
- **BitBtn2** (Yopish) tugmasi **Additional** betidan. Windows qo'llanmasining yakunlash tugmasin qaytalaydi.
- **TChart1**. Grafiglarning tuzilish komponenti **Additional** betidan olinadi.



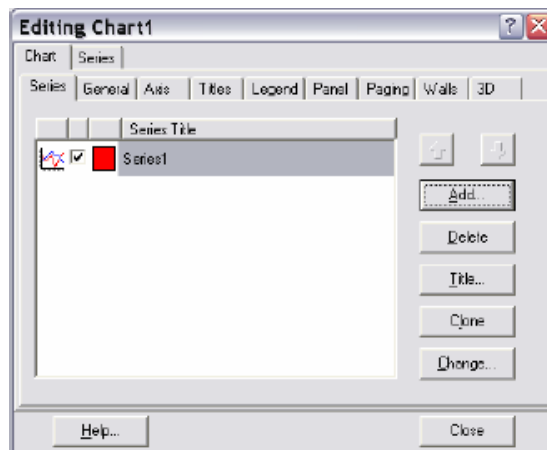
1-rasm

TChart ni ishga tayorlash uchun komponenta ustidan ikki marta bosish orqali chaqiriladigan shu komponenta xossalarining o`rnatilgan tahriri foydalaniladi. 2-rasmining chap tarafida shu tahrir oynasi ko`rsatilgan. Unda grafikni tuzich uchun **Series** zakladkasida **Add** tugmasini bosish orqali hech bo`lmaganda ma`lumotlarning bir seriyasini qo`shish kerak. Komponentaning boshqa parametrlarini o`zgarishsiz qoldirish mumkin. Bunda 2-rasmining o`ng tamonida ko`rsatilgan, grafikning tasvirlanishi usulini tanlash uchun oyna hosil bo`ladi.

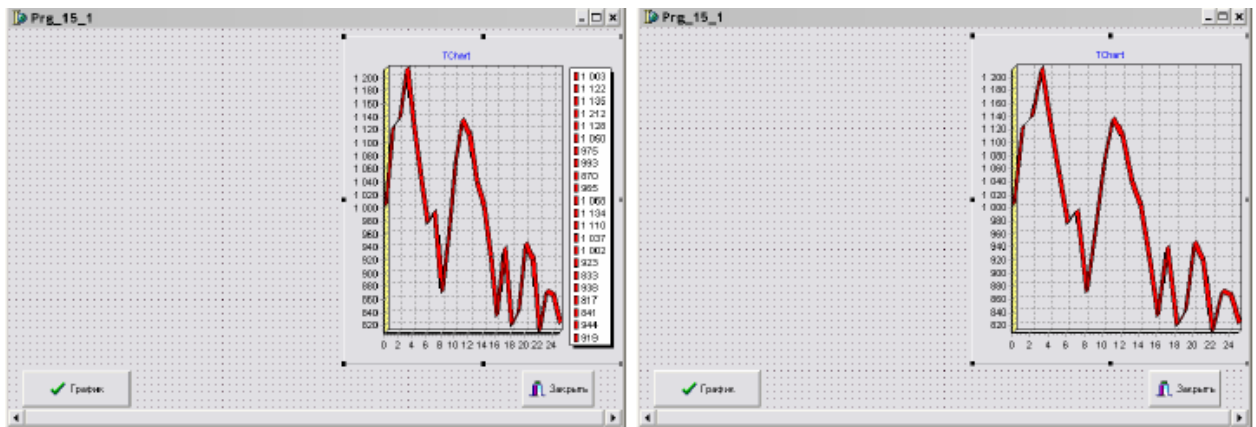


2-rasm

Grafik turi tanlagandan so`ng tahrir oynasi 3-rasmdagi ko`rinishiga ega bo`ladi, lekin **TChart** komponentida forma oynasida grafikni ko`rsatishning misol ko`rinishi ifodalanadi (3.4-rasm chap tomoniga qarang). Yana **TChart** xossalar tahririni chaqirish kerak, **Legend** vkladkasida **Visible** flagini o`shirish kerak. Forma oynasi 3.4-rasmining o`ng tamonida berilgan ko`rinishga ega bo`ladi.



3-rasm



4-rasm

Dastur listing:

```
unit Prg_3_1_;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,  
Dialogs, ExtCtrls, StdCtrls, Buttons, TeeEngine, Series, TeeProcs, Chart;
```

```
type
```

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

```
  BitBtn1: TBitBtn;
```

```
  BitBtn2: TBitBtn;
```

```
  Chart1: TChart;
```

```
  Series1: TLineSeries;
```

```
  procedure BitBtn1Click(Sender: TObject);
```

```
private
```

```
  { Private declarations }
```

```
public
```

```
  { Public declarations }
```

```

end;
var
Form1: TForm1;
implementation
{$R *.dfm}
procedure TForm1.BitBtn1Click(Sender: TObject);
var i,amp,x01,y01,x02,y02,imax,t:integer;
func:real;
begin
imax:=100;
t:=3;
amp:=70;
x01:=20;
x02:=20;
y01:=20+amp;
y02:=y01+2*amp+40;
Canvas.TextOut(0,0,'Funktsiya grafigi, Pixels');
Canvas.MoveTo(x01,y01);
Canvas.LineTo(x01+imax*t,y01);
Canvas.MoveTo(x01,y01+amp);
Canvas.LineTo(x01,y01-amp);
for i:=0 to imax*t do
begin
func:=-amp*sin(2*pi/imax*i);
Canvas.Pixels[x01+i,y01+round(func)]:=clBlack;
end;
Canvas.TextOut(x02,y02-amp-20,'Funktsiya grafigi LineTo');
Canvas.MoveTo(x02,y02);
Canvas.LineTo(x02+imax*t,y02);
Canvas.MoveTo(x02,y02+amp);
Canvas.LineTo(x02,y02-amp);
//func:=amp*sin(0);
//Canvas.MoveTo(x01,y0+round(func));
for i:=0 to imax*t do
begin
func:=-amp*sin(2*pi/imax*i);
Canvas.LineTo(x02+i,y02+round(func));
end;

```

```
for i:=0 to imax*t do
```

```
Chart1.SeriesList[0].AddXY(i,amp*sin(2*pi/imax*i),",clRed);
```

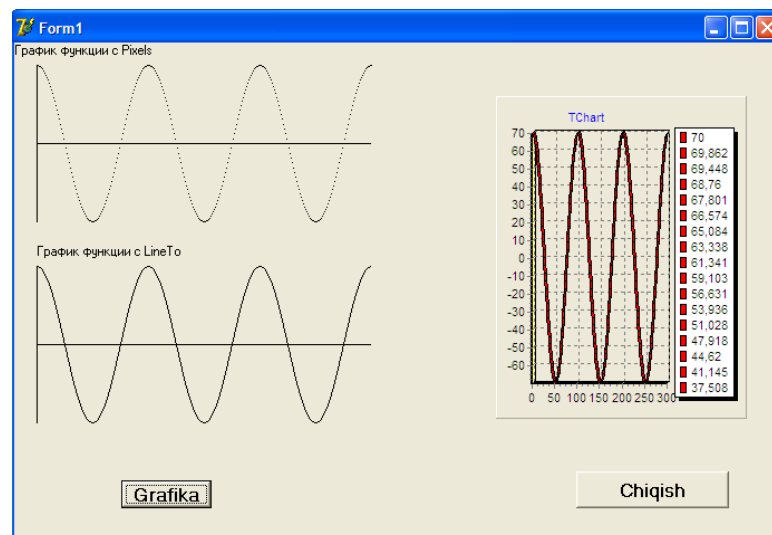
```
end;
```

```
initialization
```

```
end.
```

3.5-rasmda loyihaning Ishlab turgan oynasi ko`rsatilgan. Uning Grafik tugmasini bosganida hosil bo`ladi. Oynada bir funktsiyaning har xil usulda yaratilgan grafiklari ko`rinadi:

- **Pixels** (yuqorida chapta) xossalarin foydalanish orqali.
- **LineTo** (pasta chapta) usulin foydalanish orqali.
- **TChart** (o`ngda) komponenti yordamida.



5-rasm

3.2. Grafik ob`ektlar

Boyashga ega grafik primitiv formasidagi rasm chizish dasturin yaratish. Formada (**onDown** hodisasi) sichqon tugmasin tugmasin bosishda primitiv yaratiladi. Chiziq rangi, stili va boyash rangi sichqonning qaysi tugmasi bosilishiga bogliq (chap yoki o`ng). Loyiha-rejimi. Topshiriq variantlari 2-jadvalda keltirilgen.

2-Jadval.

1	Ellipse	clBlue	clGreen	bsSolid	clRed
2	Rectangle	clGreen	clWhite	bsHorizontal	clBlue
3	Pie	clWhite	clRed	bsVertical	clGreen
4	RoundRec	clRed	clBlue	bsDiagonal	clWhite
5	Polygon (3)	clBlue	clGreen	bsBDiagonal	clRed
6	Ellipse	clGreen	clWhite	bsCross	clBlue
7	Rectangle	clWhite	clRed	bsDiagCros	clGreen
8	Pie	clRed	clBlue	bsSolid	clWhite
9	RoundRec	clBlue	clGreen	bsHorizontal	clRed
10	Polygon (7)	clGreen	clWhite	bsVertical	clBlue
11	Ellipse	clWhite	clRed	bsDiagonal	clGreen
12	Rectangle	clRed	clBlue	bsBDiagonal	clWhite
13	Pie	clBlue	clGreen	bsCross	clRed
14	RoundRec	clGreen	clWhite	bsDiagCros	clBlue
15	Polygon (6)	clWhite	clRed	bsSolid	clGreen

Dastur bajarilishida foydalanish interfeysin o`zgartish uchun **TShape** grafikali komponentin foydalanishni ko`rsatish proektin yaratish. Loyihada o`zgaruvchi rangli formada harxil turga, o`lchamlar va ranglar bilan shakillar ifodalanadi. Topshiriq variantlari 3-jadvalda keltirilgan. Figuraning mumkin bo`lgan turlari:

3-Jadval.

Turi	Nomi
Aylana	StCircle
Ellips	stEllipse
To`rburchak	stRectangle
Burchaklari egri to`rburchak	stRoundRect
Kvadrat	stsquare

Metafayllar bilan ishlash proektin yaratish. Metafayl grafikali primitivlariga ega grafikni o`z ichiga oladi, dasturi yaratiladi, formada ifodalanadi va komp`yuterda saqlanadi. Metafaylning tug`rilig`in tekshirish uchun u formaning **Image** komponenti hisoblanadi. Loyiha – rejimida grafikali primitivlarni talaba mustaqil tanlaydi.

Bajarish Formasi.

Topshiriq 1

Bo'yashga ega grafik primitiv formasidagi rasm chizish dasturin yaratish. Primitiv – besh burchakli yulduz, formada (**onDown** hodisasi) sichqon tugmasin bosishda yaratiladi. Agar sichqonning chap tugmasi bosilgan bo`lsa, unda chiziq rang **clBlack** (qora), boyash stili **bsDiagCross** (diagonal krest), boyash rangi -

clRed (qizil). Agar sichqonning o`ng tugmasi bosilgan bo`lsa, unda chiziq rangi **clRed** (qizil), boyash stili **bsCross** (krest), boyash rangi **clBlue** (ko`k). Loyiha – rejim 3.6-rasmda berilgan.

Dastur listing:

unit Prg_3_1_;

interface

uses

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

type

TForm1 = **class**(TForm)

procedure FormMouseDown(Sender: TObject; Button: TMouseButton; Shift: TShiftState; x,y:integer);

private

{ Private declarations }

public

{ Public declarations }

end;

var

Form1: TForm1;

implementation

{ \$R *.dfm }

procedure StarLine(x0,y0,r: integer; Canvas: TCanvas);

var

p: **array**[1..11] **of** TPoint;

a: integer;

i: integer;

begin

a := 18;

for i:=1 **to** 10 **do**

begin

if (i **mod** 2 = 0)

then

begin

p[i].x := x0+Round(r/2*cos(a*2*pi/360));

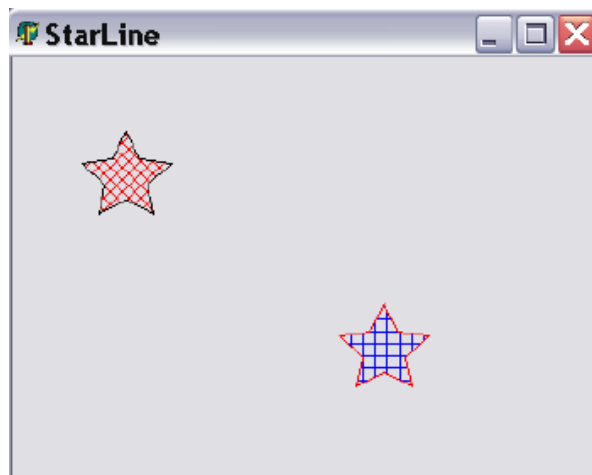
p[i].y:=y0-Round(r/2*sin(a*2*pi/360));

end

```

else
begin
p[i].x:=x0+Round(r*cos(a*2*pi/360));
p[i].y:=y0-Round(r*sin(a*2*pi/360));
end;
a := a+36;
end;
p[11].X := p[1].X;
p[11].Y := p[1].Y;
Canvas.Polyline(p);
end;
procedure TForm1.FormMouseDown(Sender: TObject; Button:TMouseButton;
Shift:
TShiftState; x,y:integer);
begin
with Form1 do
begin
if Button = mbLeft
then
begin
Canvas.Pen.Color := clBlack;
StarLine(x, y, 30,Canvas);
Canvas.Brush.Style:=bsDiagCross;
Canvas.Brush.Color:=clRed;
Canvas.FloodFill(x,y,Canvas.Pen.Color,fsBorder);
end
else
begin
Canvas.Pen.Color := clRed;
StarLine(x, y, 30,Canvas);
Canvas.Brush.Style:=bsCross;
Canvas.Brush.Color:=clBlue;
Canvas.FloodFill(x,y,Canvas.Pen.Color,fsBorder);
end;
end;
end;
end.

```



6-rasm

Topshiriq 2.

Rangi o'zgarishi mumkin bo'lgan forma turi, rangi va o'lchamin o'zgartish mumkin bo'lgan **Shape1** (shakil) komponenti tasvirlanish proektin yaratish. Shakil Formalari: **stCircle**, **stEllipse**, **stsquare**.

Bajarilishi:

1. **Delphi** ni ishga qo'shish.
2. Unda yangi loyiha rejimin yaratish.
3. **Form1** komponentining **Caption** xossasiga **Prg_3_2** ma'nosin berish.
4. Modulni **Prg_3_2_** atamasi bilan saqlash va loyihani **Prg_3_2** atamasi bilan saqlash.
5. Forma komponentalarin o'rnatish:
 - **Button1** va **Button2** tugmalari, **ComboBox1** kombinatsiyalangan ro'yxat, **ScrollBar1** va **ScrollBar2** aylantirish chizg'ishlari, **Panel1** paneli (**Standard** betidan).
 - Shakil **Shape1** (**Additional** betidan).
 - **ColorDialog1** (**Dialogs** betidan) rang dialogi.
6. **Shape1** komponentining **Shape** xossasiga **stEllipse** (Ellips) ma'nosin berish. **Brush** xossasin hosil qilib, **Shape1** ob'ektining erkin rangin o'rnatish.
7. **ScrollBar1** va **ScrollBar2** aylantirish chiziqlarining komponentlari xossalarning qiymatlarin o'rnatish: Name ScrollBar1 ScrollBar2

Kind (Tur) **sbHorizobntal** (Gorizontal) **sbVertical** (Vertikal) **Align** (Joylashtirish) **alBottom** (Pastka) **alRight** (O`ngda)

8. **Button1**, **Button2**, **ComboBox1** komponentlarin **Panel1** ob`ektining oldingi qatoriga o`rnatish va ularni vertikal buyicha tenglashtirish.

9. **Button1**, **Button2** komponentlarining xossalari qiymatlarin o`rnatish:

Button1 Button2 komponentlari

Name (Isim) **FormColor** **ShapeColor**

Caption (Nadpis) forma rangi shakil rangi

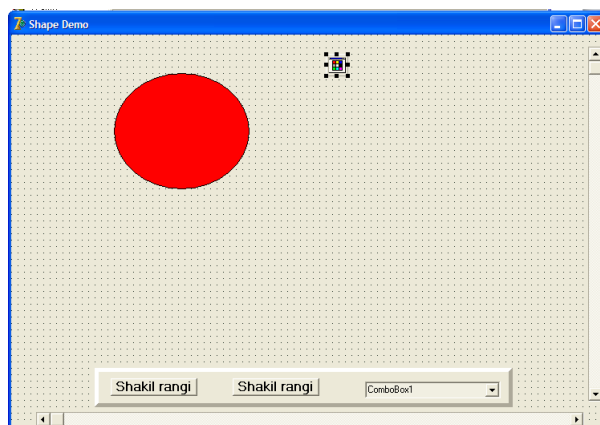
10. **ComboBox1** ro`yxatining satrlarining keyingi qiymatlarin o`rnatish:

Shakil atamasi shakil turi **stRectangle** to`g`riburchak **tSquare** kvadrat

stRoundRect egilgan burchakli to`g`riburchak **stRoundSquare** Egilgan burchakli kvadrat **stEllipse** Ellips **stCircle** Aylana

Shakillar atamalari ko`rsatilgan tartipta kiritish kerak, sababi ular satrdagi raqami buyicha analizlanadi.

11. **BorderWidth** komponentasining xossasiga **Panel1** ma`nosin berish 5. Yuqorida aytilg`an o`rnatishlardan so`ng loyiha interfeysi 7-rasmda berilgan ko`rinishga ega bo`ladi.



7-rasm

12. Loyiha rejimining bajarilish vaqtinda shakil yoki formasining rangin o`zgartish uchun rang tanlashning **ColorDialog1** standart dialog oynasin foydalanish. Bu komponent dasturning bajarilishinda ko`rinmaydi.

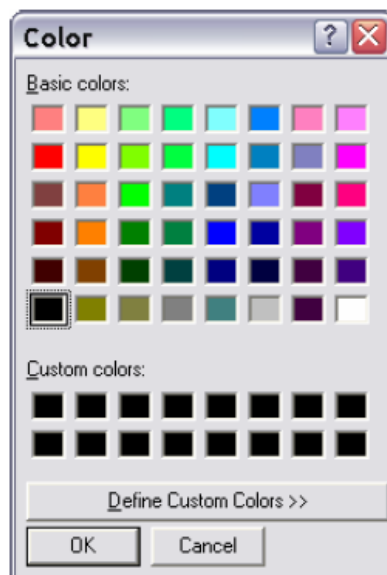
13. Tsvet formasi tugmasiga ikki marta bosish orqali Tsvet formasi tugmasi uchun **OnClick** hodisasiga javob qayta ishlashining tayorligin yaratish. Tsvet formasi

tugmasin boshganimizda **ColorDialog1** dialogli oynasi yordamida rang tanglash standart dialog oynasin chaqiradigan qayta ishlashining kodin yozish.

14. Tsvet figuri tugmasina ikki marta bosish orqali Tsvet shakil tugmasi uchun **OnClick** hodisasiga javob qayta ishlashining tayorligin yaratish.

Tugma bosilganida **ColorDialog1** dialogli oynasi yordamida tanlanadigan **Shape1** shaklining **Brush.Color** (boyash rangi) xossasi qiymatin o`rnatishi, qayta ishlashining kodin yozish.

15. Loyihani o`rinlash bo`yrug`i bilan ishga qo`shish va shakil rangi va forma rangi tugmalarining harakatin tekshirish. Boshqa oynada (8-rasm) rang tanlash tugmasin boshganida, zarur bo`lgan rangni tanlash dialog oynasida ifodalanadi.



8-rasm

Konstruktorlash rejimiga qaytish.

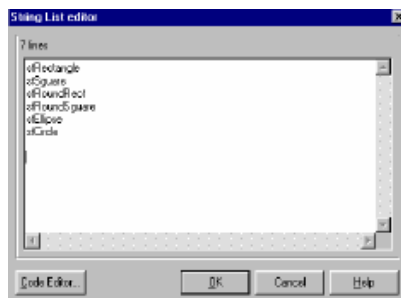
16. Aylantirish yulaklari yordamida shakilning o`lchamin o`zgartishni dasturlash. Buning uchun:

- Komponentlar ustiga ikki marta bosish orqali **ScrollBar1** va **ScrollBar2** aylantirish yulaklarining har biri uchun **OnChange** (o`zgartishga) xossasiga javob uchun tayorligin yaratish.
- Bular uchun, gorizontal (**ScrollBar1**) yoki vertikal (**ScrollBar2**) aylantirish yulaklarida “begunok” ning turgan joyiga mos turda **Shape1** shakilining **Width** (eniga) yoki **Height** (balandlik) xossasin o`rnatish kodin yozish.

- **ScrollBar1.Position** va **ScrollBar2.Position** (1...100 oraliqda yotadi) aylantirish yulaklarining “begunok” larining turgan joylari va shakillar o`lchamlari orasinga, shakilning o`lchamlarin o`zgartishing eng yaxshi ko`rinishiga erishish munosabatin tanlash.

17. Loyihani ishga qo`shish va uning aylanish yulaklarining har birining harakatin tekshirish. Konstruktorlash rejimiga qaytish.

18. Shakilning yangi turin tanlash uchun (**ComboBox**) ro`yxatin foydalanish. Loyihalash rejimida ro`yxat yaratish uchun berilgan xossasi tasvrlanishi Ob`ektlar inspektori satrining o`ng bulimida ush nuqtali kichik tugmasin bosish orqali **Items** ro`yxatiga kerakli elementlarni kiritish zarur. Ro`yxat 9-rasmda ko`rsatilgan.



9-rasm

19. Matnli tahririning dialogli oynasiga shakillarni topshiriq buyicha kiritish (9 rasmiga qarang).

20. **ComboBox1** komponenti ustida ikki marta bosish orqali **ComboBox1** ro`yxati uchun **OnClick** (bosishga) hodisalarni qayta ishlashi tayorligin yaratish. Unda ro`yxatning kerakli elementin tanlash imkoniyatin berish kodin qo`yish.

21. Kodni tekshirish uchun loyihani ishga qo`shish va keyin ro`yxatdagi shakillarning har turin ketma-ket tanlash.

22. Loyihani saqlash va yopish. **Windows** muxitidan Prg_3_3 loyihaning bajarilishi faylin ishga qo`shish va uning ishga yaroqliligini tekshirish.

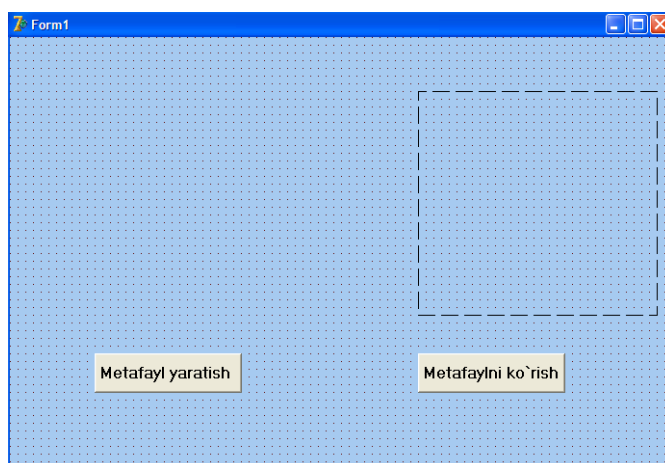
Topshiriq 3

Metafayllar bilan ishlash proektin yaratish. Metafayl ikki shakilli (ellips va kvadrat) grafikni o`z ichiga oladi, dastur yaratiladi, formada ifodalanadi va ShK diskida saqlanadi. Metafaylning to`g`riligini tekshirish uchun u formasining **Image** komponentinda saqlanadi. Loyiha – rejim 10-rasmda loyiha formasi ko`rsatilgan (3

topshiriq) . U metafaylni va ikki tugmani ko`rish uchun **Image1** komponentin o`z ichiga oladi:

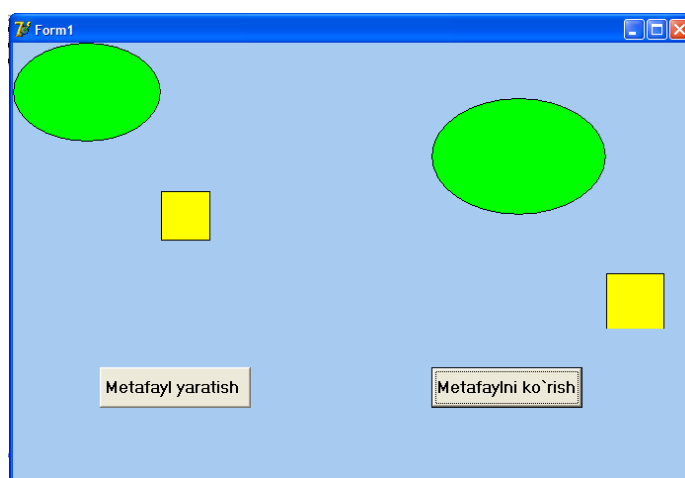
- **Button1** - «Metafayl yaratish». Metafayl tugmasin bosganimizda **MyMetafile.wmf** yaratiladi, forma yaratiladi va diskta saqlanadi.

- **Button2** - «Metafayl ko`rish». Bu tugmani bosish metafaylning formasining **Image1** komponentinda tasvirlanishiga olib keladi.



10-rasm.

Metafayl bilan ishlash dinamik xotirada amalga oshiriladi, shuning uchun yaratilgan metafaylni foydalanishdan oldin DYa ni **Free** protsedurasi bilan bo`shatish zarur. 3.11-rasmda 3 topshiriqni ishlash oynasi ko`rsatilgan. Chap bo`lagi «Metafayl yaratish» tugmasi bosilganida, o`ng bo`lagi - « Metafaylni ko`rish» tugmasi bosilganida ifodalanadi.



11-rasm

Xulosa

Ushbu bitiruv malakaviy ishida qo'yidagi natijalar olindi:

Kirish qismida mavzuning dolzarbligi asoslab berildi. Ishning birinchi bobi Ob'ektga mo'ljallangan dasturlashda tajriba mashg'ulotlarin o'tkazish asoslari atamasida bo'lib masalaning qo'yilishi, konsol va loyiha rejimlarda Delphi-7 tilida tajriba mashg'ulotlarin o'tkazish texnologiyasi ko'rsatildi va birinchi bob bo'yicha xulosa chiqarildi. Ikkinchi bob Ma'lumotlar bilan ishlash metodikasi atamasida bo'lib, unda fayllar ustida amallar bajarish, Delphi da ro'xatlar, jadvallar ustida amallar masalalari echildi. Bobning oxirida 2-bob bo'yicha xulosa chiqarildi. Uchinchi bob amaliy masalalarga tajriba mashg'ulotlarin o'tkazish metodikasi atamiasida bo'lib funktsiyalar grafigin chizish, Grafik ob'ektlar bo'yicha tajriba mashg'ulotlarin yaratish metodikasi ko'rsatildi. Bobning oxirida 3-bob buyicha xulosa keltirildi.

Bitiruv malakaviy ishining natijalarin umumiy orta maktablar, akademik lisey, kasb-hunar kollejlari va institut talabalari foydalanishlari mumkin.

Ishning oxirida foydalanilgan adabiyotlar ro'yxati va internet resurslari adresi ko'rsatilgan.

Qo'shimchalar bo'limida olingan natijalarning konsol va loyiha rejimlarining interfeyslari keltirilgan. Shuningdek, tajriba mashg'ulotlarin o'tkazishga topshiriqlar berilgan.

Adabiyotlar

1. Karimov I. Yuksak ma`naviyat – engilmas kuch. T., «Ma`naviyat», 2008.-176 b.
2. Karimov I. O`zbekiston XXI asr bo`sag`asida: xavfsizlikka tahdid, barqarorlik shartlari va taraqqiyot kafolatlari. T., «O`zbekiston», 1997. – 328 b.
- 3.Sh.Nazirov. Delphi tilida dastrurlash asoslari T.2008 y.

Elektron ta'lim resurslari

1. [http:// www.nmarket.ru](http://www.nmarket.ru)
2. Programmirovaniye v Delphi. Uchebnik. Arxangelskiy A. Ya. (www.ozon.ru/context/detail/id/2705337/)
3. [www. pedagog. uz](http://www.pedagog.uz)
4. [www. Ziyonet. uz](http://www.Ziyonet.uz)
5. [www. edu. Uz](http://www.edu.Uz)
6. <http://www.borlpasc.narod.ru/>.
7. [http:// www.intuit.ru](http://www.intuit.ru)
8. Elektronniy uchebnik po Delphi. (www.nmarket.ru/program/delphi/lessons-2).
9. Sbornik zadach po programmirovaniyu. D.M.Zlatopolskiy. <http://www.proklondike.com/contentview.php?content=516>