

O'ZBEKISTON RESPUBLIKASI
OLIY VA O'RTA-MAHSUS TA'LIM VAZIRLIGI
NAMANGAN MUHANDISLIK-PEDAGOGIKA INSTITUTI

INFORMATIKA VA AXBOROT TEXNOLOGIYALARI

fanidan

KURS ISHI

***Mavzu: Delphi dasturida ko'pburchaklarni
ifodalanishi***

***Bajardi: 9-EE-13 guruhi talabasi
Akmalov J.***

Tekshirdi: N.Niyazova

Namangan - 2014 yil.

Mundartija

Kirish

I.Asosiy qism

- 1.1.Masalani yechish uchun zarur bo'lgan boshlang'ich ma'lumotlar.....
- 1.2.Masalani qo'yilishi. Kiruvchi va chiquvchi parametrlarni aniqlash.....
- 1.3.Masalani yechishning algoritmi blok-sxemasini.....
- 1.4.Algoritmi blok-sxemasiga mos dasturning loyihasi. Delphida forma oynasini loyihalash.....
- 1.5.Dasturni kodlashtirish. Natijalar olish. Olingan natijalarni tahlil etish.....
- 1.6.Dasturni yaratish texnologiyasi va undan foydalanish yo'riqnomasi.....
- 1.7.Dasturni to'ldirish bo'yicha tavsiyalar.....
- 1.8 Qo'shimcha topshiriq uchun berilgan taqdimot sahifalarini yaratish va undan foydalanish texnologiyalarini ishlab chiqish.....

Xulosa.....

Foydalanilgan adabiyotlar

Ilova

Kirish

Kirish

Davlatimiz tomonidan keyingi paytlarda ta'lim tizimiga, ayniqsa, kollej va institutda "Informatika va axborotlar texnologiyasi" fanini kiritilishiga ahamiyat berilishi, zamonaviy kompyuterlar bilan ularni jixozlanishi o'qitish tizimida keskin o'zgarish bo'lishiga sabab bo'ldi. Xozirgi bitiruvchi yosh mutaxassislar ishlab chiqarishda to'g'ridan-to'g'ri aniq vazifalarni kompyuterda bajara olishi bunga dalil bo'ladi.

Insoniyat tarixining ko'p asrlik tajribasi ezgu g'oyalardan va sog'lom mafkuradan biron-bir jamiyatning uzoqqa bora olmasligini ko'rsatdi. Shu bois, mustaqillik tufayli mamlakatimiz o'z oldiga ozod va obod Vatan, erkin va farovon xayot barpo etish, rivojlangan mamlakatlar qatoridan o'rin olish, demokratik jamiyat qurish kabi ezgu maqsadlarni qo'ydi.

Bu esa kelajagimizni yaqqol tasavvur etish, jamiyatimizning ijtimoiy-maxnaviy poydevorini mustaxkamlash extiyojini tug'diradi. Demak, galdagi eng asosiy vazifa: yosh avlodni Vatan ravnaqi, yurt tinchligi, xalq farovonligi kabi olijanob tuyg'ular ruxida tarbiyalash, yuksak fazilatlariga ega, ezgu g'oyalar bilan qurollangan komil insonlarni voyaga yetkazish, jaxon andozalariga mos, kuchli bilimli, raqobatbardosh kadrlar tayyorlashdir.

«Jaxon sivilizatsiyasiga daxldor bo'lgan eng zamonaviy ilmlarni egallamay turib, mamalakat taraqqiyotini taxminlash qiyin», - degan edilar prezidentimiz I. Karimov. O'zbekistonning iqtisodiy va ijtimoiy soxalarda yuqori natijalarga erishishi, jaxon iqtisodiy tizimida to'laqonli sheriklik o'rnini egallay borishi, inson faoliyatining barcha jabxalarida zamonaviy axborot texnologiyalaridan yuqori darajada foydalanishning ko'lamlari qanday bo'lishiga xamda bu texnologiyalar ijtimoiy mexnat samaradorligini oshishida qanday rol o'ynashiga bog'liq.

Prezidentimiz Islom Karimovning ko'p yillik izlanishlari, asarlaridagi ikr-muloxazalariga tayanib yaratilgan «Milliy istiqlool g'oyasi: asosiy tushuncha va tamoyillar» nomli risolada ta'lim-tarbiya jarayonining ixtiyoriy bosqichida amal qilish lozim bo'lgan quyidagi mezon va talablar keltirilgan:

- o'quv mashg'ulotlarini olib borishda talabalarning yoshi, tafakkuri, dunyoqarashi va qiziqishlarini xisobga olish;
- taxlim-tarbiyaning ilg'or, taxsirchan vositalaridan, zamonaviy o'qitish texnologiyasi imkoniyatlaridan keng foydalanish;
- ayrim tushunchalarni xaddan ziyod soddalashtirish, taxlimning eskicha uslub va tamoyillarini qo'llash natijasida fanning qadrsizlanishiga yo'l qo'ymaslik;
- taxlim jarayonida tazyiq o'tkazmasdan ma'rifiy asosda ish tutish, yoshlarning mustaqil va erkin fikrlash, baxs-munozara yuritish ko'nikmalarini oshirishga e'tibor qaratish;
- o'qituvchi va tinglovchilar orasida o'zaro xamfikirlik va xamkorlik muxitini shakllantirish, mavzuning tushuncha va tamoyillarini sharxlashda xayotiy misollar, bugungi dunyoda ro'y berayotgan voqealar taxlilidan, matbuot materiallaridan keng foydalanish.

Yuqoridagi fikrlardan kelib chiqib, bu – kurs ish «Ekranida gorizontal xarakat qilayotgan ko'pburchak» tasvirini yaratishga bag'ishlangan.

U quyidagi qismlardan iborat:

- kirish;
- asosiy qism;
- xulosa;
- foydalanilgan adabiyotlar;
- ilovalar.

Asosiy qism

1.1. Masalani yechish uchun zarur bo'lgan boshlang'ich ma'lumotlar

Kurs ishini yozish va bajarishda quyidagi adabiyotlar foydalandim va foydalanilgan adabiyotlar ro'yxati quyida keltirilgan.

Mualliflar professorlar. Sh.Nazirov, M.Musaev va dotsentlar A.Ne'matov, R.Qobulovlar tomonidan yozilgan "Delphi tilida dasturlash asoslari" nomli o'quv qo'llanmada Delphi tilining asosiy tushunchalari, operatorlari, Delphi muhiti bilan ishlash, Delphi muhitida grafika va multimediya, Delphi qo'shimcha imkoniyatlari, ma'lumotlar bazasining nazariy asoslari va Delphi vizual dasturlash vositasi yordamida ma'lumotlar bazasini yaratish texnologiyalari haqida ma'lumotlar keltirilgan.

Mualliflar t.f.n., dotsent Fayzullaev S.X., assistent Akramov A.A.lar tomonidan yozilgan "Informatika va AT" fanidan yozilgan o'quv qo'llanmada zamonaviy komp yuter va axborot texnologiyalari asoslari, operatsion tizimlar, matn va grafik muharrirlari, elektron jadvallar, tasvirlarga ishlov berish usullari, komp yuter tarmoqlari, axborotlarni izlanish tizimi, Internet hamda elektron pochta haqida talabalarga mukammal bilim berish, yuqori darajadagi algoritmik tilda turli dasturlar tuzishga hamda mutaxassis-texnolog sifatida amaliyotda vujudga kelgan turli muammolarni komp yuterda tez va aniq bajarishga o'rgatish masalalari o'rganilgan. Ushbu o'quv qo'llanma talabalarni hozirgi zamonaviy axborot texnologiyalari bo'yicha o'z bilimlarini yanada mustahkamlashga va o'quv-uslubiy jarayonlarda qo'llashga mo'ljallangan. (www.ziyo.net.)

Buxoro oziq ovqat va engil sanoat texnologiyasi instituti «Informatika va axborot texnologiyalari» kafedrasida katta o'qituvchisi Narziev U.Z. tomonidan «Informatika va axborot texnologiyalari» fanidan yozilgan ma'ruzalar matni 5140900 Kasb ta'limi "Informatika va axborot texnologiyalari" ta'lim yo'nalishi talabalariga «Ob'ektga mo'ljallangan dasturlash» fanidan ma'ruzalarni o'tkazish uchun mo'ljallangan. Ma'ruzalar matnida Delphi dasturlash tili asosiy tushunchalari, loyihani boshqarish, komponentlar to'plami sharhi, multimedia dasturlarini yaratish vositalari, matnli va grafik chop qilish, Delphi ob'ektlarning xususiyatlari, ma'lumotlar ombori bilan ishlash asoslari masalalari o'rganilgan. (www.ziyo.net.)

Namangan muhandislik-pedagogika instituti "Informatika va AT" kafedrasida dotsent S.Irisqulov, katta o'qituvchi M.Ataxonov, katta o'qituvchi T.Jo'raevlar tomonidan yozilgan muammoli ma'ruzalar matnida masalani EHMda yechish bosqichlari, Delphi tilining asosiy tushunchalari, operatorlari, Delphi muhiti bilan ishlash va uning qo'shimcha imkoniyatlar beruvchi komponentlari haqida ma'lumotlar keltirilgan. (S.Irisqulov, M.Ataxonov, T.Jo'raev. Vizual dasturlash tilidan muammoli ma'ruzalar matni. Namangan-2010y.)

Mualliflar T.Xolmatov, N.Taylaqov, U.Nazarovlar tomonidan yozilgan "Informatika" nomli darslikda Hisoblash texnikasi rivojlanish bosqichlari, MS DOS operatsion tizimi, NC qobiq dasturi, Windows operatsion tizimi, Microsoft Word va Excel dasturlari, Paint grafik muharriri, Microsoft Access dasturi, Internet Explorer va elektron pochta xizmatidan foydalanish hamda algoritmik asoslari va Paskal algoritmik tili haqida ma'lumotlar keltirilgan. (T.X.Xolmatov,

1.2. Masalani qo'yilishi. Kiruvchi va chiquvchi parametrlarni aniqlash.

Kurs ishini bajarish mobaynida men imkoniyat va bilim darajamdan kelib chiqib "Ekranda gorizontaal xarakat qilayotgan ko'pburchak dasturini yaratish" masalasini tanlab oldim va shu mavzu asosida kurs ishi masalasini xal etmoqchiman. Ushbu mavzuni bajarishda Delphi dasturining grafik imkoniyatlaridan foydalandim.

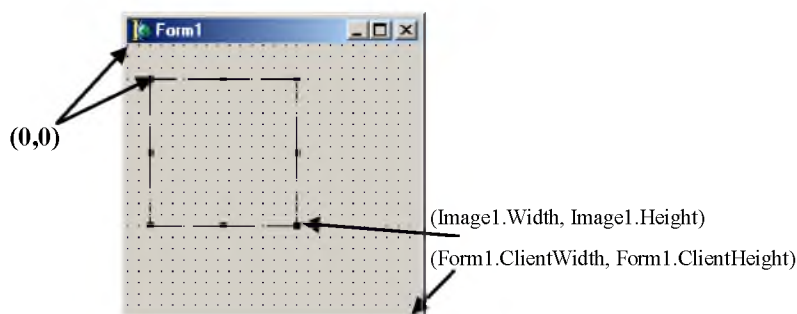
Delphining grafik imkoniyatlari.

Delphi dasturchiga turli xildagi sxemalar, chizmalar va illyustrasiyalar bilan ishlash imkoniyatlarini beradi. Dastur grafikani ob'ekt (forma yoki komponent Image) sirtida hosil qiladi. Ob'ekt sirti Canvas xususiyatiga mos keladi. Grafik element (to'g'ri chiziq, aylana, to'g'ri to'rtburchak va x.k.)larni ob'ekt yuzasida hosil qilish uchun Canvas dan foydalaniladi.

Masalan, **Form1.Canvas.Rectangle(10,10,50,50)** instruksiyasi dastur oynasida to'g'ri to'rtburchak hosil qiladi.

Chizma hosil bo'luvchi sirt.

Yuqorida aytib o'tilganidek, grafikani hosil qiluvchi sirt (yuza) Canvas xususiyatiga to'g'ri keladi. O'z navbatida Canvas xususiyati TCanvas tipidagi ob'ektdir. Bu tip uslublari grafik primitivlarni (nuqta, chiziq, aylana va x.k.) hosil bo'lishini ta'minlaydi, xususiyati esa hosil bo'luvchi grafikani xarakteristikalarini: rangi, chiziq qalinligi va turi; bo'yaluvchi hududni rangi va ko'rinishini; harfni xarakteristikalarini beradi. Canvas «sirt», «chizish uchun yuza» sifatida tarjima qilinadi. Chizish yuzasi alohida nuqta – piksellardan tashkil topadi. Pikselni joylashuvi gorizontaal (X) va vertikal (Y) koordinatalar bilan xarakterlanadi. CHap yuqoridagi nuqta koordinatasi (0,0). Koordinatalar yuqoridan pastga va chapdan o'ngga qarab o'sib boradi.



Chizish yuzasi nuqta koordinatalari.

CHizish yuzasi o'lchamlarini illyustrasiya (Image) hududi uchun Height va Width, forma uchun esa ClientHeight va ClientWidth lar aniqlash mumkin.

Qalam va mo'yqalam.

Odatda rassom surat chizish uchun qalam va mo'yqalamdan foydalanadi. Delphi ning grafik imkoniyatlari ham qalam va mo'yqalamdan foydalanish imkoniyatlarini yaratadi. Qalamdan chiziq va kontur chizishda, mo'yqalamdan esa kontur bilan chegaralangan yuzani bo'yash uchun foydalaniladi.

Qalam va mo'yqalam grafikani chizish yuzasida hosil qilishda mos ravishda Pen(qalam) va Brush(mo'yqalam) xususiyatlariga xosdir. SHu bilan birga ular TPen va TBrush tiplariga tegishlidir.

Qalam.

Qalamdan nuqta, chiziq, geometrik shakllar: to'g'ri to'rtburchak, aylana, ellips va h.k. larni chizishda qurol sifatida foydalaniladi. TPen ob'ekt xususiyati 1-jadvalda keltirilgan.

1-jadval

Xususiyat	Vazifasi
Color	CHiziq (kontur) rangi
Width	CHiziq qalinligi
Style	CHiziq ko'rinishi
Mode	Tasvirlash rejimi

Color xususiyati chizuvchi qalam rangini belgilaydi. Quyidagi 2-jadvalda PenColor xususiyatlari keltirilgan:

2-jadval

Konstanta	Rang	Konstanta	Rang
clBlack	qora	clSilver	kumushrang
clMaroon	kashtanrang	clRed	qizil
clGreen	yashil	clLime	salatrang
clOlive	olivkoviy	clBlue	ko'k
clNavy	to'q ko'k	clFuchsia	Fuchsia
clPurple	atirgulrang	clAqua	yorug' ko'k
clTeal	Teal	clWhite	oq
clGray	kulrang		

Width xususiyati chizuvchi qalam qalinligini (pikselda) belgilaydi.

Masalan, Canvas.Pen.Width:=2 chiziq qalinligi 2 pikselga teng bo'ladi.

Style xususiyati chiziluvchi chiziqning turini belgilaydi. Style komponentlari 3-jadvalda keltirilgan.

3-jadval

Konstanta	CHiziq ko'rinishi
psSolid	To'g'ri chiziq
psDash	Uzun shtrixli punktir chiziq
psDot	Qisqa shtrixli punktir chiziq
psDashDot	Uzun-qisqa shtrixli punktir chiziq
PsDashDotDot	Bir uzun va ikki qisqa shtrixli punktir chiziq
PSClear	Ko'rinmas chiziq

Mo'yqalam.

Mo'yqalam (Canvas.Brush)dan yopiq sohalarni to'ldirish uchun foydalaniladi, masalan, geometrik shakllarni bo'yash va x.k. Mo'yqalam ob'ekt sifatida quyidagi ikki xususiyatni o'z ichiga oladi:

Color – bo'yaluvchi soha rangi

Style – to'ldiruvchi soha tipi

Masalan, konturning ichki sohasi bo'yalishi yoki shtrixlanishi mumkin.

Color xususiyati sifatida Tcolor ning barcha o'zgarmlaridan foydalanish mumkin. Style xususiyatlari 4-jadvalda keltirilgan.

4-jadval

Konstanta	Bo'yaluvchi soha tipi
bsSolid	to'liq
bsClear	Bo'yalmaydi
bsHorizontal	gorizontal shtrixlash
bsVertical	vertikal shtrixlash
bsFDDiagonal	oldinga egilgan diagonal shtrixlash
bsBDDiagonal	orqaga egilgan diagonal shtrixlash
bsCross	gorizontal-vertikal setkali shtrixlash
bsDiagCross	diagonal setkali shtrixlash

Matn hosil qilish.

Grafik ob'ekt sirtida matnni hosil qilish uchun TextOut dan foydalaniladi. TextOut ning yozilish formati quyidagicha:

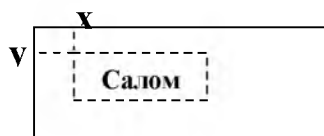
Ob'ekt.Canvas.TextOut(x,y,Text);

Bu yerda

Ob'ekt – matn hosil bo'luvchi ob'ekt nomi;

x,y – matn boshlanuvchi koordinata (2-rasm);

Text – hosil bo'luvchi belgi kattalikdagi matn yoki satrli o'zgaruvchi.



2-rasm. Matn hosil bo'luvchi soha koordinatasi

Hosil bo'luvchi matn belgilari Canvas ob'ektiga muvofiq keluvchi Font xususiyati orqali ifodalanadi. Font xususiyati TFont ob'ektiga tegishli bo'lib, 5-jadvalda belgi xarakteristikalarini va qo'llaniluvchi uslublarini keltirilgan.

5-jadval

Xususiyat	Aniqlanishi
Name	Foydalaniluvchi shrift. Qiymat sifatida shrift nomi yoziladi, masalan, Arial Cyr
Size	punktlarda ifodalaniluvchi shrift o'lchami. Punkt-poligrafiyada qo'llaniluvchi o'lchov birligi bo'lib, u taxminan 1/72 dyuym ¹ ga teng
Style	belgini yozish usuli, quyidagicha bo'lishi mumkin: oddiy, qalin, kursiv, ostiga chizilgan, ustiga chizilgan. Bular quyidagi konstantalar yordamida amalga oshiriladi: <i>fsBold</i> (qalin), <i>fsItalic</i> (kursiv), <i>fsUnderline</i> (ostiga chizilgan), <i>fsStrikeOut</i> (ustiga chizilgan). style bir nechta usullarni kombinasiya qilishi mumkin. Masalan, qalin kursiv holatini ifodalash: <i>Ob'ekt.Canvas.Font:=[fsBold, fsItalic]</i>
Color	Belgi rangi. Qiymat sifatida <i>TSolor</i> konstantalaridan foydalanish mumkin.

Quyidagi dastur qismi TextOut funksiyasini qo'llash uchun misol bo'la oladi:

with Form1.Canvas **do**

begin

```
Brush.Color:=Form1.Color;  
Font.Size:=14;  
Font.Style:=[fsItalic, fsBold];  
TextOut(10,10,'Salom, Delphi!');
```

End;

Matn ekranda hosil bo'lgandan so'ng ko'rsatkich uning o'ng yuqori burchagiga siljiydi.

Ba'zida matndan so'ng biror ma'lumotni chiqarish kerak bo'lib qoladi. Agar matn uzunligi noma'lum bo'lsa ko'rsatkich turgan koordinatani aniqlash mushkul. Masalan «so'm» so'zini raqamdan keyin hosil qilish kerak bo'lsin. Bunday holatlarda ko'rsatkich turgan koordinatadan boshlab davom etish uchun PenPos dan foydalanishga to'g'ri keladi:

```
TextOut(10,10,SumPr); // SumPr – String tipli kattalik  
TextOut(PenPos.X, PenPos.Y, ' sum');
```

Ko'pburchak

Polygon uslubi yordamida ko'pburchak chiziladi. Parametr sifatida Tpoint tipli massivni qabul qiladi. Massivning xar bir elementi yozuvni saqlaydi. Bu yozuv (x,y) maydon bo'lib, ko'pburchak uchlarining koordinatalaridir. Polygon uslubi massivda joylashgan x,y koordinatalarni ketma-ket to'g'ri chiziqni tutashtiradi va oxirgi koordinata bilan birinchisini xam to'g'ri chiziq birlashtiradi.

Kontur chiziqlarining ko'rinishi (rangi, qalinligi va stili) Pe xususiyati bilan aniqlanadi, soxa ichini rangi va stili esa Brush xususiyati bilan aniqlanadi.

Quyidagi protsedurada Polygon uslubidan foydalanib uchburchak chizish tashkil qilingan.

```
procedure TForm1.Button2Click(Sender: TObject);  
var  
pol: array[1..3] of TPoint; // uchburchak uchlarining koordinatalari  
begin  
pol[1].x := 10;  
pol[1].y := 50;  
pol[2].x := 40;  
pol[2].y := 10;  
pol[3].x := 70;  
pol[3].y := 50;  
Form1.Canvas.Polygon(pol);  
end;
```

1.3. Masalani yechishning algoritm blok-sxemasi

Algoritmni ishlab chiqish.

Bu bosqichda yechilayotgan masalaning natijasini olish uchun bajarish lozim bo'lgan amallar ketma ketligini aniqlash zarur bo'ladi. Agar masala bir nechta usullar bilan hal qilinishi, yoki natijalarning bir necha variantlarda olish mumkin bo'lsa, dasturchi biror bir kriteriyaga (masalan, algoritmning bajarilish tezligiga) asoslangan sodda, eng maqbul yechim yoqi usulni tanlaydi. Algoritmni ishlab chiqish natijasida masala yechish yo'lining so'zlar yoqi blok-sxema orqali yozilgan batafsil ifodasi (algoritmi) hosil qilinadi.

Dastur yaratishning birinchi bosqichida dasturchi quyilgan masalani to'la hal qilish uchun bajarilishi zarur bo'lgan amallar ketma-ketligi va undagi tartibni aniqlaydi, ya'ni algoritm quradi. **Algoritm** echilayotgan masala doirasida boshlangich ma'lumotlardan natijaga o'tish jarayonini ifodalovchi aniq, qo'rsatmalardir.

Ta'rif: Algoritm deb qo'yilgan masalani to'la xal uchun bajarilishi zarur bulgan amallar ketma-ketligining qat'iy tartibiga aytiladi.

Masalaning yechish algoritmi so'zlar orqali, maxsus matematik formulalar yoki maxsus blok-sxema deb ataluvchi maxsus grafiq ko'rinishda ifodalanishi mumkin.

Shuningdek, ixtiyoriy dorilarni tayyorlash yo'llari, ovqatlarni tayyorlash usullari, xakim belgilagan dorilarni iste'mol qilish, bankomatdan pul olish kabi amallarni algoritm sifatida qabul qilish mumkin. Algoritmarga xayotiy va turli fan soxalaridagi masalalarni yechish yo'llari xam kiradi.

Algoritmarga quyidagi talablar qo'yiladi :

1. Boshlanishi va tugashi qo'rsatilishi kerak.
2. Xar qanday amal buyruq, tarzida ifodalanishi shart.
3. Xar bir amal ijrochiga tushunarli bo'lgan qo'rinishda ifodalangan bo'lishi shart.
4. Har bir amalda qatnashayotgan o'zgaruvchilarning qiymatlari oldindan aniqlangan bo'lishi kerak.
5. Xar qanday amal natijasi bir qiymatli bo'lishi kerak.
6. Bajariladigan amallar soni cheqlangan bo'lishi kerak.
7. Yaquniy natijalarni ajratib qo'rsatish va chiqarish shart.
8. Masalani to'la yechish uchun berilgan xamma ma'lumotlar va mumkin bo'lgan barcha imkoniyatlar xisobga olingan bo'lishi kerak.
9. Algoritm ommaviy, ya'ni bitta sinfga taalukli bo'lgan ko'plab masalalarni yechishga mo'ljallangan bo'lishi kerak.

Yuqoridagi talablarning birortasi buzilgan bo'lsa, qo'yilgan masalani yechish uchun qurilgan algoritm to'la qonli bo'la olmaydi, ya'ni masalaning to'la yechimini bera olmaydi. Masalan: Agar biron bir amalni bajarishda qatnashayotgan har bir o'zgaruvchining qiymati oldindan aniqlanmagan (4-talab) bo'lsa, u holda ana shu o'zgaruvchining o'rniga odatda 0 qo'yib hisoblanadi. Bu esa har doim ham to'g'ri natija beravermaydi. Faraz qilayliq, K -o'zgaruvchining qiymati oldindan aniqlanmagan bo'lsin. U holda $D = (A + B)/K$ ifodaning qiymatini hisoblashning iloji yo'q, chunki K ning o'rniga qompilyator nol qiymatini qo'yadi. Natijada nolga bo'linish holati ro'y beradi. Bunday bo'lishi esa mumkin emas. Endi 6-talabni buzib ko'raylik.

1. *Hisoblansin* $i = 1$;
2. *Hisoblansin* $i = i + 1$;
3. *1-ga o'tilsin.*

Bu holda qurilgan algoritm «cheqsiz algoritm» bo'lib qoladi, yani uni «ijrochi» hech qachon tugata olmaydi.

Blok-sxemalar usulida algoritmning xar bir buyrugi maxsus geometriq shaqlar yordamida ifodalanadi.

1.4. Algoritm blok-sxemasiga mos dasturning loyihasi. Delphida forma oynasini

loyihalash

Formalar bilan ishlash

Delphida tuziladigan dasturlar biror forma asosida qilinadi. Delphida har bir yangi hosil qilingan formaga unga mos bo'lgan modul avtomatik tashkil qilinib turiladi. Bu esa dasturchi uchun juda qulay imkoniyat, yani uning ishini tezlashtirishga yordam beradi.

Bu formaning shaklini tanlash, unda komponentalarni joylashtirish bizning ixtiyorimizda bo'ladi. Formaga biror komponentani qo'ymoqchi bulsak, shu komponentaning ustida sichqonchaning chap tugmasini ikki marta bosiladi. Bu komponenta formaning o'rtasiga joylashadi. Biz uni xoxlagan joyimizga surib ko'chirishimiz mumkin.

Komponentalar.

Matnli komponentalar. Bu komponentalar ma'lumotlarni kiritish, saqlash, tahrir qilish va turli xabarlar berish vositasi sifatida foydalaniladi.



-Tlabel (belgi) komponentasi.

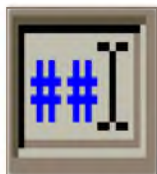
Belgi bu sizning dasturingizdagi boshqa ob'ektlarni identifikatsiyalashganligi uchun foydaliroqdir. Belgi sizga mo'ljallashtirishda va bundan tashqari foydalanuvchiga o'z nomidan tashqari axborot berishda hamda oyna va belgilarni tavsiflashda yordam beradi.

WordWrap va AutoSize xossalari. Belgining ushbu xossasi belgi komponentasini o'rnatishda foydaliroqdir. Word Wrap qiymati true bo'lsa, Delphi aftomatik ravishda qatorni tamg'ani nomlanishiga o'tkazadi. Auto Size xossasining qiymati true ga teng bo'lsa bu uning asl qiymatidir (po umolchaniya).

OnClick hodisasi. Bu hodisa foydalanuvchiga dasturni mo'ljallashtirishda yordam berishi uchun foydali hisoblanadi.



EDIT komponentasi orqali matnli ma'lumotlarni kiritish va keyinchalik dastur jarayonida bu qiymatdan foydalanish imkonini beradi.



MaskEdit ko'p hollarda ma'lumotlarni yashirish holda yoki biror bir kodlash orqali kiritishga to'g'ri kelib qoladi. Shunday holda bu komponentadan foydalanish mumkin. Masalan parol kiritishda va hokazo.



Memo komponentasi sodda matn muhariri bo'lib, u ko'p satrli ma'lumotlarni kiritish va chiqarish uchun ishlatiladi.

Bu komponentalar mantlar ustida amallar bajarish uchun ishlatiladi.

Password Char xossasi. Ushbu xossa matnni kiritishdan ximoya uchun foydalaniladi. Parol kiritish kerak paytida foydalaniladi. Ushbu xossadan siz maxfiylikning qo'shimcha darajasini ta'minlash uchun va foydalanuvchi ismini yashirish uchun foydalanish mumkin.

Read Only xossasi. Ushbu xossa ob'ektning qiymatini taxrirlash imkoniyatini boshqaradi. Siz uning qiymatini False yoki True qilib qo'yishingiz mumkin. Agarda xossaning qiymati True

bo'lsa foydalanuvchi ushbu ob'ektdan foydalanishga, ya'ni ixtiyoriy matnni belgilash va nusxasini buferga olish uchun birmuncha ruxsat beradi.

Max Length xossasi. Ushbu xossadan uchala ob'ektga kiritiladigan simvollarni cheklash uchun foydalaniladi.

Mask Edit komponentasining umumiy xossalaridan tashqari qo'shimcha xossalarga ham egadir. Ushbu xossadan bu ob'ekt orqali klaviaturadan kiritiladigan simvollarni filterlash uchun foydalaniladi. Maska qanday simvol qabul qilinishi yoki qilinmasligini aniqlaydi.



-List Box komponentasi.

Ushbu komponenta foydalanuvchiga tanlash uchun elementlar ro'yxatini ko'rsatadi. Ayrim payti siz o'z dasturingizda juda ko'p elementlarning ro'yxatini ko'rsatish lozim bo'lib qoladi. Bunday xolatlar uchun ro'yxatlar darchasida standart ko'rinishdagi surgich paydo bo'ladi va yordam beradi. Ro'yxatlar paneliga elementlarni qo'shish yoki olib tashlash mumkin. Ushbu komponentaning ichki ko'rinishini o'zgartirish uchun mo'ljallangan turli xil xossalari mavjud.

Ro'yxatlar darchasini formada joylash uchun ko'p joy zarur bo'ladi. Kombinatsiyalashgan panel komponentasi **Combo Box** ni formada joydan ekonom qilish uchun foydalanish qulayroqdir.

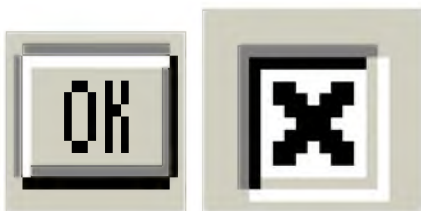


-Combo Box komponentasi.

Combo Box komponentasi **Sombination Boxes** so'zidan qisqartirilgan bo'lib **List Box** komponentasiga o'xshashdir. Lekin kombinatsiyalashgan darcha kam joyni talab qiladi va bir vaqtning o'zida bitta elementni ko'rsatadi.

Demak **List Box** komponentasi bilan **Combo Box** komponentasining farqi bir vaqtning o'zida qancha element ko'rsata olishda ekan. Joydan tejash maqsadida **Combo Box** dan foydalanish qulayroq bo'lsa, ko'proq elementni ko'rsatish uchun **List Box** dan foydalanish qulay.

Button va Check Box komponentalari.



Button (tugma) va **Check Box** komponentalarining ko'pgina Windows muhitida uchratish mumkin.

Button (tugma) komponentasi.

Bu komponenta odatda dialog darchalarida foydalaniladi. Ko'pgina Windows dasturlarida tanish bo'lgan **Ok** va **Cancel** juftligini ko'rgansiz. Ammo sizga tugma komponentasidan foydalanish uchun dialog darchasini yaratish shart emas. Tugma komponentasini siz dasturingizning istalgan joyida joylashtirishingiz mumkin.



-BitBtn komponentasi.

Grafik tugma komponentasi oddiy tugma komponentasiga judaham o'xshashdir. **Delphi** tugmalarning standart majmuasini taklif qiladi. Bularni dasturingizda yaratish uchun tanlab olishingiz mumkin.

Chek Box (belgilash tugmasi) komponentasi ushbu tugma odatda dasturda «**Xa**» yoki «**Yo'q**» javobini olish kerak bo'lgan vaziyatlarda foydalaniladi. Demak bu komponentada ortiqcha bitga xolatga ega bo'ladi.

Grafik komponentalar

Grafik komponentalar formada figura va ob'ektlarni ko'rsatish hamda ma'lumotlarni akslantirish uchun foydalaniladi. Bu komponentalarni ba'zilar juda ham oddiy ba'zilar juda ham murakkab. Ularni har biri anik vazifa uchun mo'ljallangan.



Image komponentasi

Bu komponentani formada rasmlarni joylashtirish uchun foydalaniladi. Rasmni nomi esa **Picture** xossasida keltiriladi. Bu rasmlarning kengaytmasi **.VMR, ICO, WMF** bo'lishi mumkin. Agar siz **Image** komponentasining o'lchamini o'zgartirishni xohlasangiz **Auto Size** xossasiga **True** qiymati berishingiz lozim. **Stretch** xossasiga **True** berilgan bo'lsa, belgilangan sohaga joylashtiriladi.



Paint Box komponentasi. (chizish darchasi)

Ushbu komponenta formaning cheklangan qismida chizish imkonini beradi. Chizish darchasi o'zining chegaralarini ko'rsata olmaydi. Ushbu qiyinchilikni yo'qotish uchun uning ichiga

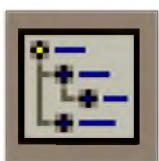
guruxlashgan panel **Croup Boxes** qo'yish va uni o'lchamini **Pait Box** komponenta o'lchami bilan bir xil qilish lozim. Ana shundagina sizga chizish uchun ramka paydo bo'ladi.



Shapes (figuralar) komponentasi

Figuralar komponentasi o'z tabiati bilan judayam oddiy. Siz sodda geometrik figuralardan dasturingizni stillashtirish foydalanishingiz mumkin.

Figuralarning va konturlarning rangi mos ravishda **Pen** va **Brush** xossalari yordamida o'rnatiladi.



Outlines (sxemalar) komponentasi

Sxemalar isxarxik tarzda tashkil qilingan ma'lumotlarni akslantirish uchun foydalaniladi. Misol uchun katologlar daraxti.

Sxema **Lines** va **Items** xossalari orqali murojat qilish mumkin bo'lgan qatorlardan tashkil topadi.

Outlines Tile (sxemali) xossasi istalgan konfiguratsiyasini olish uchun foydalaniladi.



Color Crids komponentasi (ranglar to'ri)

Bu komponenta chizish va fon ranglarini boshqarish uchun interfeysni ifodalaydi. Bu komponenta yordamida chizish va fon rangini tanlashda foydalaniladi.

Foreground (chizish rangli) va **Background** (fon rangi) xossalarda belgilangan rangdan tashkil topgan bo'ladi.

Fayl va katalogga murojat qilish komponentalari

Ushbu komponentalar fayl va kataloglar bilan turli amallarni bajarishda foydalaniladi.



File List Box (fayllar ro'yxati paneli) komponent fayllar ro'yxati paneli

komponentasi katalogga murojat interfeysini ifodalaydi.

Mask xossasi filter yoki maskani o'rnatish uchun foydalaniladi.

Foydalanuvchi tomonidan belgilangan fayl nomi **File Name** xossasi orqli murojaat qilishingiz mumkin.

1.5. Dasturni kodlashtirish. Natijalar olish. Olingan natijalarni tahlil etish.

Dastur kodi: Dastur kodi forma orqasiga yashiringan bo'lib, u yerga dastur matnlari kiritiladi. U oynaga F12 yoki Ctrl+F12 tugmalari yordamida o'tish mumkin.

Delphi kodlar muhitida dastur buyruqlarini kiritish va ularni qayta ishlash mumkin. Shuni ham ta'kidlash lozimki, Delphi kodlar muhiti avtomatik tarzda Object Pascal dasturlash tilidagi kalit so'zlarni (begin, end, procedure, const, var va bosh.) qalin harflar bilan belgilaydi.

Ma'lumot yozilgan satr(dastur izohi)ni belgilash uchun figurali qavslardan foydalaniladi. Qavs ochilsa undan keyin turgan kodlar ko'rinishi o'zgaradi. Kerakli joyda qavs berkitilsa ko'rinishi o'zgargan kodlar faqat qavs oralig'idagina qoladi va dastur ishlash jarayonida shu oraliq ishlatilmaydi.

Delphi kodlar muhitining imkoniyatlaridan yana biri shuki, u yerga biror funksiyani masalan: «**StrToFloat**» ni yozib, qavs ochsak satr ostida kichik oyna hosil bo'ladi. Bu oynada qavs ichidagi o'zgaruvchi tipi ko'rsatilgan bo'ladi, yoki biror operatorni masalan: **Label1** ni yozib nuqta qo'yilsa satr ostida nuqtadan keyinga yozish mumkin bo'lgan operatorlar ro'yhati chiqadi va ulardan kerakligini tanlab qo'yishimiz mumkin.

Kodlar oynasida biror operator sirtiga kursorni olib borib Ctrl+F1 tugmalari teng bosilsa shu operator haqidagi yordam oynasi hosil bo'ladi. U yerdan kerakli axborotni olish mumkin. Agar kursorni bo'sh joyga olib kelib, F1 bosilsa umumiy yordam ma'lumotlari chiqadi.

Kodlar oynasida tahrirlash oddiy matn muharrirlari kabi amalga oshiriladi. Ya'ni belgilangan (blokka olingan) kod nusxasini olish, qirqib olish va kerakli joyga qo'yish mumkin. Undan tashqari kodlar ichidan kerakli belgini izlab topish va almashtirish, **Delete** tugmasi yordamida kursordan keyin turgan belgini, **Backspace** yordamida esa kursordan oldin turgan belgi yoki belgilarni o'chirish mumkin. **Ctrl+→**, **Ctrl+←** tugmachalari yordamida bir so'z keyinga va oldinga, **PgDn**, **PgUp** tugmachalari yordamida esa bir oyna pastga va yuqoriga o'tiladi.

Dastur bajarilayotganda yuz beradigan xatoliklar.

Odatda dastur tuzilayotganda ba'zi kamchilik yoki xatoliklarga yo'l qo'yamiz, dasturni ishga tushirgan vaqtimizda esa bu xatoliklar to'g'risida axborot beruvchi oyna hosil bo'ladi va bu xatoliklar quyidagilarga bo'linadi: Sintaktik; Dastur bajarilish vaqtidagi xatoliklar; Algoritmik.

Sintaktik xatoliklar operator, prosedura-funksiya, o'zgaruvchi nomi va boshqalarni notug'ri yozishdan kelib chiqadi.

Sintaktik xatoliklarni kompilyatsiya vaqtida chiquvchi xatoliklar ham deb yuritiladi (Compile-time error). Bu xatoliklarni topish boshqa xatoliklarni topishga nisbatan ancha yengil hisoblandi. Ularni kompilyator topadi va kursoni ana shu xatolik sirtiga olib kelib qo'yadi, dasturchi esa uni to'g'rilab, dasturni boshqattan ishga yuklaydi.

Dastur bajarilish vaqtidagi xatoliklar dastur ishga yuklanganidan so'ng yoki uni Testdan o'tkazayotgan vaqtda xosil bo'ladi. Masalan bu xatoliklar, massiv chegarasidan chiqib ketish, proseduraga yoki funksiyaga noto'g'ri bog'lanish, fayllar bilan noto'g'ri ishlash va boshqalar. Xatolik yuzaga kelgan vaqtda dastur o'z ishini to'xtatadi va Delphi **“Error”** oynasida (xatolik) xatolik xaqidagi ma'lumotlarni chiqaruvchi dialogli oyna xosil bo'ladi. Bu oynadagi ma'lumotdan kerakli xulosa chiqariladi.

Dastur ishini davom ettirish uchun **Error** dialogli oynadagi **OK** tugmachasi bosiladi, so'ngra **Run** menyusidan **Program Reset** buyrug'i tanlanadi.

Algoritmik xatoliklarni aniqlash ko'pchilik hollarda qiyinchilik tug'dirishi mumkin. Dastur ishga yuklanganida xatolik xaqida xech qanday ma'lumot oynaga chiqarilmaydi, lekin natija noto'g'ri chiqadi. Bu turdagi xatoliklarni aniqlash uchun dasturchi dasturni qadamlab bajartirishi va har bir qadamdagi natijani tekshirib borsa maqsadga muvofiq bo'ladi.

Dasturni qadamlab bajartirish uchun birinchi navbatda **Run** menyusidagi **Add watch (Ctrl+F5)** buyrug'i tanlanadi va o'zgaruvchi qiymatlarini o'zida saqlaydigan (**Watch List**) oyna xosil bo'ladi. Unga xatolikni tekshirish jarayonida zarur bo'lgan o'zgaruvchilar kiritiladi. So'ngra kursor noto'g'ri bajarilayotgan algoritm sirtiga olib boriladi va **Run** menyusidan **Run to Cursor (F4)** buyrug'i tanlanadi. Yuqoridagi amalni bajarganimizdan so'ng, dastur ishga yuklanadi va dasturdagi ketma-ketlik kursor turgan joyga kelganida dastur o'z ishini vaqtinchalik to'xtatadi. Dasturning qolgan qismini **F7** tugmachasi yordamida qadamlab bajartirish mumkin va har bir qadamdan so'ng, **Watch List** oynasidagi o'zgaruvchi qiymatlari tekshirib boriladi.

1.6. Dasturni yaratish texnologiyasi va undan foydalanish yo'riqnomasi.

Berilgan masala uchun Dastur loyihasini keltirib o'tamiz:

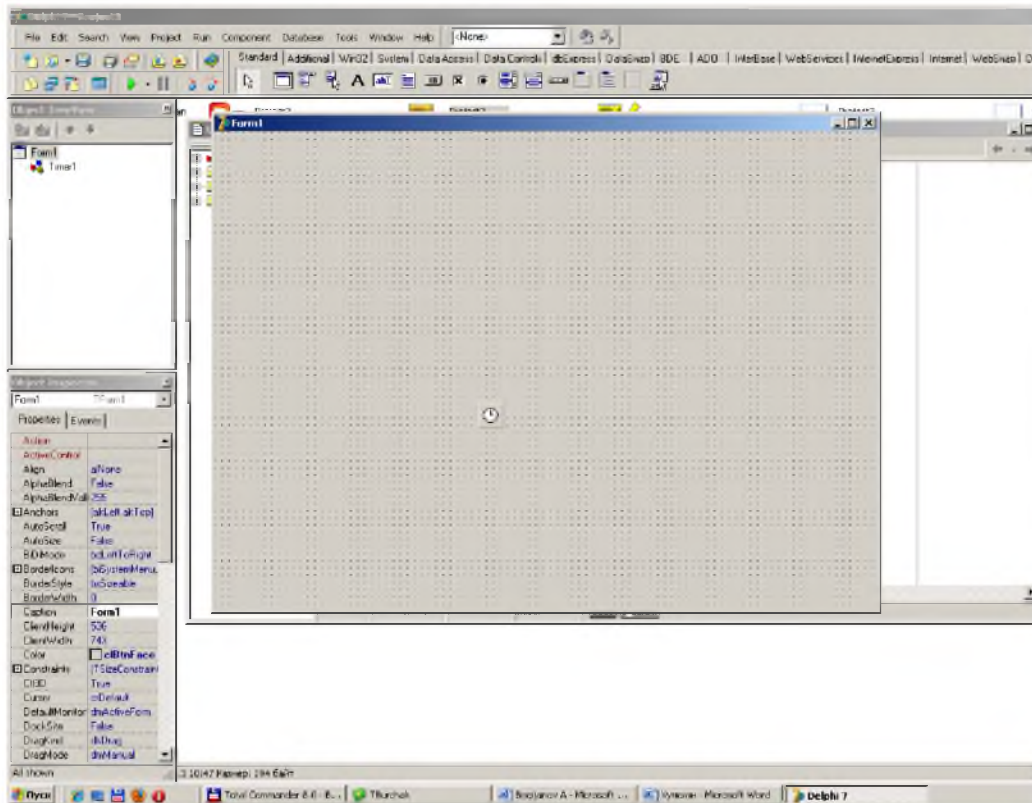
Forma oynasiga 1 ta Timer (taymer) dan foydalanamiz. Timer yordamida harakatni vaqt bo'yicha amalga oshirilishi ta'minlangan. Timer komponenti komponentlar palitrasining System bo'limida joylashgan. Timer xususiyatlari quyidagi jadvalda keltirilgan.

Konstanta	Bo'yaluvchi soxa tipi
Name	Komponent nomi
Interval	Millisekundlarda beriluvchi On Timer generatsiyasi
Enabled	Ishga ruxsar berish. Qiymat True bo'lsa ruxsat beriladi, false bo'lsa berilmaydi.

On Timer xodisasi Timer komponentini ishga tushiradi. **On Timer** vaqtli xodisasi millisekundlarda o'zgaradi va **Interval** xususiyatlariga mos keladi. **Enabled**

xususiyati esa dasturda **timerni** “ishga tushirish” yoki “to’xtatish” imkoniyatini yaratadi. Agar **Enabled** True(rost) bo’lsa **OnTimer** xodisasi ishlamaydi.

Natijada ekranda dastur loyihasi hosil bo’ladi:



So’ngra Timer tugmasini bosish (**Click**) hodisasi uchun va kodlar oynasidagi Dastur ko’rinishini keltirib o’tamiz:

```
unit Unit1010;
```

```
interface
```

```
uses
```

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

```
type
```

```
TForm1 = class(TForm)
  Timer1: TTimer;
  procedure Timer1Timer(Sender: TObject);
  procedure FormCreate(Sender: TObject);
```

```
private
  { Private declarations }
public
  { Public declarations }
end;
```

```
var
  Form1: TForm1;
  x,y,dx:integer;
  pol:array[1..6] of TPoint;
```

implementation

```
{ $R *.dfm }

  Procedure Ris;
  begin
    //aylana ko'rinmas qilish
    form1.Canvas.Pen.Color:=form1.Color;
    //    form1.Canvas.Rectangle(x,y,x+100,y+100);
    y:=y+dx;// aylanani yangi joyda xosil qilish
    pol[1].X:=x+100;  pol[1].y:=y+100-dx;
    pol[2].X:=x+200;  pol[2].y:=y+150-dx;
    pol[3].X:=x+300;  pol[3].y:=y+150-dx;
    pol[4].X:=x+400;  pol[4].y:=y+100-dx;
    pol[5].X:=x+300;  pol[5].y:=y+50-dx;
    pol[6].X:=x+200;  pol[6].y:=y+50-dx;
    //pol[7].X:=10;    pol[7].X:=10;
```

```
form1.Canvas.Polygon(pol);
```

```
    //x:=x+dx;
```

```
    form1.Canvas.Pen.Color:=ClBlack;
```

```
//    form1.Canvas.Rectangle(x,y,x+100,y+100);
```

```
pol[1].X:=x+100; pol[1].y:=y+100;
```

```
pol[2].X:=x+200; pol[2].y:=y+150;
```

```
pol[3].X:=x+300; pol[3].y:=y+150;
```

```
pol[4].X:=x+400; pol[4].y:=y+100;
```

```
pol[5].X:=x+300; pol[5].y:=y+50;
```

```
pol[6].X:=x+200; pol[6].y:=y+50;
```

```
//pol[7].X:=10;    pol[7].X:=10;
```

```
form1.Canvas.Polygon(pol);
```

```
end;
```

```
procedure TForm1.Timer1Timer(Sender: TObject);
```

```
begin
```

```
ris;
```

```
end;
```

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

```
begin
```

```
    x:=0;
```

```
    y:=10;
```

```
    dx:=5;
```

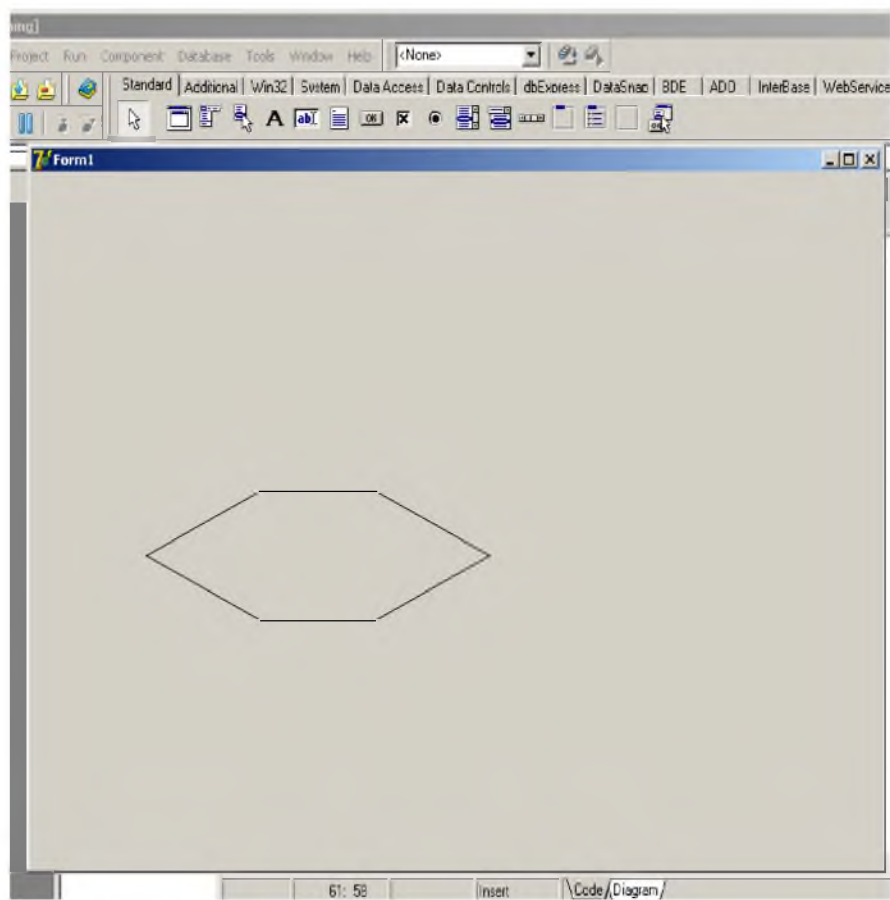
```
    Timer1.Interval:=50;
```

```
    Form1.Canvas.Brush.Color:=Form1.Color;
```

```
end;
```

```
end.
```

Dastur matni kiritilgach, dasturni ishga tusgirish uchun F9(**Run**) – tugmasi bosiladi.



1.7.Dasturni to'ldirish bo'yicha tavsiyalar.

Delphi dasturlash tili bir qancha imkoniyatlarga ega. Delphining imkoniyatlaridan foydalangan xolda yana quyidagi ishlarni amalgam oshirish mumkin: Grafik imkoniyatlaridan foydalangan xolda rasmlar bilan axborot almashish mumkin, video tasvirlarni qayta ishlash mumkin, operatyion tizimlar bilan qo'shimcha dastur tayyorlash mumkin, ba'zalar bilan ishlash mumkin.

1.8 .Qo'shimcha topshiriq uchun berilgan taqdimot sahifalarini yaratish va undan foydalanish texnologiyalarini ishlab chiqish.

Power Point dasturi xaqida tushuncha

Shaxsiy komp yuterlarni ishlab chiqarishga va ayniksa maorif sohasiga kirib kelishi o'kuvchilarga darslarni komp yuter imkoniyatlaridan foydalanib o'rganishlariga katta yo'l ochib bermokda. Komp yuterdagi xilma-xil dasturlar foydalanuvchilarga xech bir malakali o'rgatuvchi yoki tayyorlovchisiz o'zi mustaqil holda shu mavzu to'g'risida ma'lumot olishga kattta yordam beradi. Bu dasturlardan foydalanib katta-katta zallarda seminarlar, konferentsiyalar o'tkazish mumkin. Endi komp yuter dasturlari imkoniyatlaridan foydalanib, bir xonadagi komp yuterda o'tirib, bir nechta auditoriyalarda bir vaqtning o'zida ko'plab tinglovchilarga axborot yetkazish mumkin. Agar komp yuter Internet tarmog'iga ulangan bo'lsa, boshqa mamlakatlardagi komp yuter va komp yuter zallarida seminar va konferentsiyalar o'tkazish imkoniyati ochiladi. Biz ushbu mashg'ulotda axborotlarni tinglovchilarga taqdim etish vositasi bo'lgan PowerPoint dasturi bilan tanishamiz. PowerPoint dasturi Microsoft kompaniyasining maxsulotidir. U Windows operatsion tizimida ishlashga mo'ljallangan. PowerPoint dasturi xam Microsoft Office dasturlar paketiga kiradi. Bu dasturdan foydalanib, biz xujjatlarni animatsiya shaklida foydalanuvchiga yetkazishimiz mumkin.

Animatsiya - bu xujjatlarni ham matn, ham tovush, xam videotasvir holatida ifodalanishidir. Bu bilan biz foydalanuvchiga shu xujjat to'g'risida to'laroq va mukammalroq axborot berishimiz mumkin. Power Point dasturida yaratilgan xujjatlarni prezentatsiya deb ataladi.

Prezentatsiya – bu biror bir ma'lumotni matn, jadval, diagramma va animatsiya shaklida foydalanuvchiga taqdim etishdir. Prezentatsiya ichida rasm va videotasvirlarni ham ishlatish mumkin. Bu dastur yordamida yaratilgan fayllarni kengaytmasi .ppt shaklida bo'ladi. Matn ham, tovush ham, animatsiya ham birgalikda bir fayl ichida saqlanadi. Prezentatsiyalar o'z ichiga slaydlarni oladi. Slaydlar ichida matn yoki animatsiyalar bo'lgan prezentatsiyaning bir qismidir. Slaydlarning barchasi prezentatsiyani tashkil kiladi. Slaydlar soni cheklanmagan. Bir slayddan qandaydir bir effekt bilan ikkinchi slaydga o'tiladi. Slayddan slaydga o'tish uchun sichqoncha tugmasi bosiladi. Avvalgi yoki keyingi slaydga o'tish uchun yo'nalish tugmalaridan foydalaniladi.

PowerPoint dasturini ishga tushirishni 3 xil usulda amalga oshirish mumkin:

1. «Pusk» tugmachasini bosib, bosh menyuga chikiladi. Bosh menyuda «Programm» menyusi tanlanadi. Bu menyu ichida PowerPoint -  Microsoft PowerPoint piktogrammasida sichqonchanning chap tugmasi bosiladi.

2. Microsoft Office panelida  - PowerPoint piktogrammasi aktivlashtiriladi (sichqoncha chap tugmasi bosiladi).

3. Biror bir .ppt kengaytmali fayl yoki prezentatsiyani ishga tushirish orqali. Bunda PowerPoint dasturi avtomatik tarzda yuklanadi va oynaga ushbu fayl yoki prezentatsiya chiqadi.

Xulosa

Xulosa.

Xulosa qilib shu aytishim mumkinki, men o'zimga berilgan ushbu kurs ishini bajarish mobaynida quyidagi natijalarga erishdim.

1. Delphi dasturlash tilini mustaxkam o'rgandim;
2. Delphi dasturlash tilining grafik imkoniyatlaridan foydalangan xolda dastur tuzishni va dasturdan natija olishni;
3. Qo'shimcha topshiriqni bajarish mobaynida Power Point dasturini yanada mustaxkamladim.

Kurs ishini bajarish davomida bir qancha natijalarga erishish bilan birgalikda, bir qancha sintaktik, bajarish vaqtidagi xatolik va algoritmik xatoliklarga duch keldim.

Sintaktik xatoliklar nisbatan oson to'g'rilanadi. Ularni kompilyator aniqlaydi, men esa faqat dastur matniga o'zgartirish kiritdim xolos.

Bajarish vaqtidagi xatoliklar ham, Delphida ular mustasnolar (exception) deb ataladi, qoidaga asosan oson to'g'rilanadi. Ular oddatda dasturni birinchi ishga tushirganda va testlash vaqtida paydo bo'ladi.

Xatolik sodir bo'lgandan keyin dasturning ishini to'xtatish uchun **Run** buyrug'ini tanlash kerak.

Foydalanilgan adabiyotlar

Foydalanilgan adabiyotlar

1. T. Xolmatov, N.I.Taylaqov. «Amaliy matematika, dasturlash va komp`yuterning dasturiy ta`minoti», Mehnat - Toshkent , 2000 y.
2. N.Otaxanov. “Delphi”da dasturlash. 2005 y.
3. M.Aripov, B.Begalov, Sh.Begimqulov «Axborot texnologiyalari», T.: Noshir, 2009 y»,»,
4. Jo`raev T. «Borland Delphi 7» (o`quv qo`llanma) Namangan, 2012 y. 45-50 bet,114-125 bet.
5. «Texnologik mashina va jixozlarni avtomatlashtirish» fanidan ma`ruzalar matni.
6. Internet ma`lumotlarini olish mumkin bo`lgan saytlar:
www.ziynet.uz
www.nammpi.uz;
www.gov.uz.
www.referat.uz. ,
www.Google.uz ,
hisoblash@rambler.ru

Ilova

Ilova

```
unit Unit1010;
```

```
interface
```

```
uses
```

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

```
Dialogs, ExtCtrls;
```

```
type
```

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

```
    Timer1: TTimer;
```

```
    procedure Timer1Timer(Sender: TObject);
```

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

```
  private
```

```
    { Private declarations }
```

```
  public
```

```
    { Public declarations }
```

```
end;
```

```
var
```

```
  Form1: TForm1;
```

```
  x,y,dx:integer;
```

```
  pol:array[1..6] of TPoint;
```

```
implementation
```

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

```
  Procedure Ris;
```

```
  begin
```

```
    form1.Canvas.Pen.Color:=form1.Color;
```

```
    y:=y+dx;// ko'pburchakni yangi joyda xosil qilish
```

```
    pol[1].X:=x+100; pol[1].y:=y+100-dx;
```

```
    pol[2].X:=x+200; pol[2].y:=y+150-dx;
```

```
    pol[3].X:=x+300; pol[3].y:=y+150-dx;
```

```
    pol[4].X:=x+400; pol[4].y:=y+100-dx;
```

```
    pol[5].X:=x+300; pol[5].y:=y+50-dx;
```

```
    pol[6].X:=x+200; pol[6].y:=y+50-dx;
```

```
    form1.Canvas.Polygon(pol);
```

```
    form1.Canvas.Pen.Color:=ClBlack;
```

```
    pol[1].X:=x+100; pol[1].y:=y+100;
```

```
    pol[2].X:=x+200; pol[2].y:=y+150;
```

```
    pol[3].X:=x+300; pol[3].y:=y+150;
```

```
pol[4].X:=x+400; pol[4].y:=y+100;
pol[5].X:=x+300; pol[5].y:=y+50;
pol[6].X:=x+200; pol[6].y:=y+50;
form1.Canvas.Polygon(pol);
    end;
procedure TForm1.Timer1Timer(Sender: TObject);
begin
    ris;
    end;
procedure TForm1.FormCreate(Sender: TObject);
begin
    x:=0;
    y:=10;
    dx:=5;
    Timer1.Interval:=50;
    Form1.Canvas.Brush.Color:=Form1.Color;
    end;
end.
```