

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

INFORMATIKA VA AXBOROT TEXNOLOGIYALARI

fanidan

KURS ISHI

***Mavzu: Delphi dasturida harakat shakllarini
ifodalanishi***

***Bajardi: 9-EE-13 guruhi talabasi
Abdullayev A.***

Tekshirdi: N.Niyazova

Namangan - 2014 yil.

Mundarija

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 algoritm blok-sxemasini.....
- 1.4. Algoritm 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

Ma'lumki, axborot-kommunikatsiya texnologiyalari davlat va nodavlat sektorlari o'rtasida o'zaro muomala qilish va axborot ayirboshlashning printsiptal jihatdan yangi shakllari va imkoniyatlarini ochadi, fuqarolik jamiyati barpo etilishi va mustahkamlanishiga ko'maklashadi, iqtisodiy islohotlar va mamlakatni demokratlashtirish jarayonlarini yanada jadallashtirish imkonini beradi.

Darhaqiqat, uquv jarayonlarida axborot-kommunikatsiya texnologiyalarini joriy etish va rivojlantirish jamiyatni axborot jihatidan keng ko'lamli ta'minlashning ajralmas qismi hisoblanadi. Ushbu omil davlat boshqaruvining samaradorligini jiddiy ravishda oshirish, davlatning biznes tuzilmalari va fuqarolik jamiyati institutlari bilan o'zaro munosabatlarini maqbullashtirish imkonini beradi.

Kelajagi buyuk davlat kurish tafakkuri, dunyokarashi uzgargan yoshlarimiz, mutaxassis kadrlarimizga kup jixatdan boglikdir. Shuning uchun yangicha fikrlaydigan, bozor iqtisodiyoti sharoitida muvoffakiyatli xujalik yuritadigan yuksak malakali, chukur bilimli mutaxassislarni tayyorlash davr talabi bulib koladi.

Rivojlangan jamiyatda axborotni kayta ishlashning asosiy texnik vositasi sifatida shaxsiy komp yuterlar ishtirok etadilar. Xozirgi kunda mamlakatimizning rivojlangan davlatlar katorida salmokli mustaxkam urin egallashi uchun zamonaviy komp yuter texnologiyalari jamiyatimizning barcha tabaqalarida, shu jumladan, ta'lim tmzimidagi isloxatlarni amalga oshirishda keng va atroflicha joriy etish maksadga muvofikdir. Biz axborot olami tarakkiyotidagi uzgarishlar jadal rivojlanishlarning guvoxi bulayapmiz va yangi axborot texnologiyalari yuzaga kelmokda.

Respublikamizda olib borilayotgan islohotlarning amalga oshirilishida yuqori malakali mutaxassislarning roli benihoya kattadir. Prezidentimiz ta'kidlaganlaridek: «Ertangi kun yangicha fikrlay oladigan, zamonaviy bilimga ega bo'lgan yuksak malakali mutaxassislarni talab etadi». Shu sababli xalqimizning boy intellektual merosi va umumbashariy qadriyatlari, zamonaviy madaniyat, iqtisodiyot, fan, texnika va

texnologiyalar asosida yetuk mutaxassislar tayyorlash tizimi ishlab chiqildi va jadal sur'atlar bilan xayotga tatbiq etilmoqda.

Ta'lim mazmunini isloh qilib, uni jahon ta'lim standartlariga muvofiqlashtirishda ilg'or pedagogik texnologiyalarni joriy etishga alohida e'tibor berilgan. Xususan, Kadrlar tayyorlash milliy dasturida «o'quv jarayonini ilg'or pedagogik texnologiyalar bilan ta'minlash» zarurligi ham alohida ko'rsatib o'tilgan.

Xozirgi paytda respublikamizda yosh avlodni tarbiyalash, ukitish, bilim berish, zamonaviy informatsion texnologiyalarga yakindan yondashish, xamda yangi texnika va texnologiyalar bilan ishlashni urganishga sharoit yaratish maksadida juda kup amaliy ishlarni amalga oshirib borilmokda. Shu amalga oshirilib borilayotgan ishlarni asosiylaridan biri o'rgatuvchi elektron pedagoglar yaratish va ularni dars jarayonlariga tadbiq etish texnologiyalarini ishlab chiqishdir. Chunki ta'limning bu turi shu paytgacha mavjud bulgan ta'lim turlaridan uzining ayrim ijobiy tomonlari bilan ajralib turadi. Mazkur ta'lim turiga juda keng axoli ommasini jalb etish mumkin.

Yuqoridagi fikrlardan kelib chiqib, bu – kurs ishi massiv kattaliklarga doir masalalar yechish dastur ta'minotini yaratishga bag'ishlangan.

U quyidagi qismlardan iborat:

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

Kirish qismida hozirgi kunda axborot kommunikation texnologiyalari, Infomatika va AT fanini o'qitish axamiyati va dolzarbligi qisqacha keltirilgan.

Asosiy qismda kurs ishini xal qilinishi lozim bo'lgan masalalar, masalaning qo'yilishi va uning uslubiy ahamiyati kabilar keltirib o'tilgan.

Foydalanilgan adabiyotlar qismida shu kurs ishni tayyorlash davomida foydalanilgan adabiyotlar, foydalnilgan veb-saytlar ro'yxatlari keltirilgan. Ilovalar qismida dastur kodlari bayon etilgan.

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, Delfida ob'ektlarning xususiyatlari, ma'lumotlar ombori bilan ishlash asoslari masalalari o'rganilgan. (www.ziyo.net.)

Namangan muhandislik-pedagogika instituti “Informatika va AT ” kafedrası dotsent S.Irisqulov, katta o’qituvchi M.Ataxonov, katta o’qituvchi T.Jo’raevlar tomonidan yozilgan muammoli ma’ruzalar matnida masalani EHMda yechish bosqichlari, Delfi tilining asosiy tushunchalari, operatorlari, Delfi 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 bosqich-lari, 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 foy-dalanish hamda algoritmlash asoslari va Paskal algoritmik tili haqida ma’lumotlar keltirilgan. (T.X.Xolmatov, N.I.Tayloqov, U.A.Nazarov. Informatika. Toshkent, O’zbekiston milliy entsiklopediyasi, 2003y.- 256 b.)

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

Delphi dasturlash tili uchun Paskal dasturlash tili asos qilib olingan. Bu ikki tilning qanday o'xshashlik va farqli tomonlari bor?

Bilamizki, Paskalda biror kattaroq dastur, masalan amaliy dasturlar majmuini tuzmoqchi bo'lsak, albatta biz modulli dasturlashdan foydalanamiz. Ya'ni qo'yilgan masalani kichikroq bo'laklarga bo'lib olamiz va shular bilan ishlaymiz. Natijani olish uchun esa, bu bo'laklarni birlashtiruvchi bitta asosiy dastur qilinadi. Aynan shu narsa Delphi dasturlash tilining asosi xisoblanadi. Agar biz Delphida biror forma hosil qilsak, uholda Delphi bizga o'zi avtomatik tarzda shu formaga mos bo'lgan modullarni va bu modullarni boshqaruvchi dasturni tuzib beradi. Boshqaruvchi dasturni Delphida, agar unga o'zimiz nom bermasak, Project1.dpr deb nomlaydi. Bunda .dpr fayl kengaytmasi. Avtomatik tarzda tashkil qilingan modulga biz nom bermasak, u holda Delphi uni Unit1.pas deb nomlaydi. O'zimiz esa .pas kengaytmasini saqlagan holda hohlagan nomimizni berishimiz mumkin.

Tuzilgan dasturni EXM xotirasida saqlash uchun asosiy menyuning File bandidan foydalaniladi. Dasturni kompilyatsiya qilish uchun asosiy menyuning RUN bandidan foydalaniladi. Agar dasturda xatoliklar bo'lsa, kompilyatsiya qilish jarayonida bu xatoliklar ko'rsatib boriladi. Agar xatolik bo'lmasa, dastur ishga tushadi.

Ob'ektlar inspektoridan foydalanish

Windows uchun tuzilgan programmaning ichki tuzilishi konsol programmalarning ichki tuzilishidan farqlanadi. Dastur bajarilishida kalit so'zlar bo'lgan begin va end orasiga olingan operatorlarni windows boshqacha shaklda boshqaradi, yani ularni hodisa deb qaraydi. Har bir hodisaning o'ziga mos xossalari mavjud. Dasturlashda bu xossalardan keragini tanlab ishlatiladi. Bu tanlash ob'ektlar inspektorida qilinadi. Ob'ektlar inspektori ikki qismdan iborat Properties yani xodisalar va Events xossalar. Delphi ning xarakterli tomoni shundan iboratki, agar biz biror kompanentadan foydalansak, ob'ektlar inspektori unga mos xodisalarni ajratib ko'rsatib turadi. Masalan biz biror yangi forma xosil qilgan bo'lsak, ob'ektlar

inspektorida captionda Form1 yozuvi turadi. Agar biz xoxlasak Form1ning o'rniga yangi o'zimizning programmaga mos nomimizni berishimiz mumkin. Har bir komponentaga obektlar inspektorida unga tegishli xodisa va shu xodisaga mos xossalar bo'ladi.

Delphi dasturlash tilining yana bir xarakterli tomoni shundan iboratki, agar biz dasturda biror tugmaga qandaydir vazifani yuklamoqchi bo'lsak, shu tugmaning ustida sichqonchani ikki marta chertilsa, shu tugmaga mos keluvchi protseduraning ichiga avtomatik ravishda kiritib qo'yadi. Bundan tashqari dastur tuzish jarayonida formadan modulning ichiga va moduldan formaga o'tishga extiyoj bo'ladi, buning uchun asboblari panelidan Toggle Form-Unit tugmasi tanlanadi yoki F12 tugmasini bosish bilan amalga oshiriladi.

Multiplikatsiya.

Multiplikatsiya deyilganda odatda harakatlanuvchi yoki o'zgaruvchi rasmni tushuniladi. Oddiy holatlarda rasm harakatlanishi yoki o'zgarishi mumkin. Hosil qilingan rasm (chiziq, aylana, yoy va x.k.)larni siljitish juda oddiy: avval rasm hosil qilinadi, bir ozdan so'ng uni tozalanadi va yana yangitdan avvalgi joyidan boshqa yerda hosil qilinadi. Bunday almashtirish bir maromda davom ettirilsa, natijada tasvir oyna bo'ylab harakatlanayotganga o'xshaydi.

Quyidagi kichik dastur yordamida aylanani dastur oynasining chap chegarasidan o'ng chegarasiga qarab harakatlantirishimiz mumkin. Rasmda forma ko'rinishi keltirilgan.



Dasturi

unit Unit1;

interface

uses

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

type

TForm1 = **class**(TForm)

Timer1: TTimer;

procedure Timer1Timer(Sender: TObject);

procedure FormActivate(Sender: TObject);

private

{ Private declarations }

public

{ Public declarations }

end;

var

Form1: TForm1;

x, y, dx: byte;

implementation

{ \$R *.dfm }

procedure Ris;

begin

{ aylanani ko`rinmas qilish }

Form1.Canvas.Pen.Color := form1.Color;

Form1.Canvas.Ellipse(x, y, x + 10, y + 10);

X := x + dx;

```

{aylanani yangi joyda xosil kilish}
Form1.Canvas.Pen.Color := clBlack;
Form1.Canvas.Ellipse(x, y, x + 10, y + 10);
end;
procedure TForm1.Timer1Timer(Sender: TObject);
begin
    ris;
end;
procedure TForm1.FormActivate(Sender: TObject);
begin
    x := 0;
    y := 10;
    dx := 5;
    Timer1.Interval := 50;
    Form1.Canvas.Brush.Color := Form1.Color;
end;
end.

```

Asosiy ishni aylanani o'chirib yangi joyda hosil qiluvchi Ris prosedurasi bajaradi. Aylanani o'chirishni uning rangini forma rangiga o'zgartirish yo'li bilan amalga oshiriladi.



Timer

Forma yoki dasturda e'tibor bergan bo'lsangiz vizual bo'lmagan komponent **Timer** (taymer)dan foydalandik. Uning 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 soha tipi
<i>Name</i>	Komponent nomi
<i>Interval</i>	Millisekundlarda beriluvchi OnTimer generatsiyasi
<i>Enabled</i>	Ishga ruhsat berish. Qiymat true bo'lsa ruhsat beriladi, false bo'lsa berilmaydi.

OnTimer xodisasi **Timer** komponentini ishga tushiradi. **OnTimer** vaqtli xodisasi millisekundlarda o'zgaradi va **Interval** xususiyatlariga mos keladi. **Enabled** xususiyati esa dasturda taymerni «ishga tushirish» yoki «to'xtatish» imkoniyatini yaratadi. Agar **Enabled True** (rost) bo'lsa **OnTimer** xodisasi ishlamaydi.

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 yo'qi 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 aniqdangan 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.

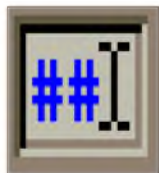
Belgi bu sizning dasturingizdagi boshqa ob'ektlarni identifikatsiyalashganligi uchun foydaliroqdir. Belgi sizga mo'ljallashtirishda va bundan tashqari foydalanuvchiga o'z nomidan tashqari axborot berishda hamda oyna va belgilarni tavsiflashda yordam beradi.

WordWrap va AutoSize xossalari. Belgining ushbu xossasi belgi komponentasini o'rnatishda foydaliroqdir. **Word Wrap** qiymati **true** bo'lsa, **Delphi** avtomatik ravishda qatorni tang'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.

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 xar 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 xoxlasangiz **Auto Size** xossasiga **True** qiymati berishingiz lozim. **Stretch** xossasiga **True** berilgan bo'lsa, belgilangan sohaga joylashtiriladi.



Pait Box komponentasi. (chizish darchasi)

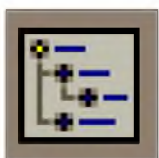
Ushbu komponenta formaning cheklangan qismida chizish imkonini beradi. Chizish darchasi o'zining chegaralarini ko'rsata olmaydi. Ushbu qiyinchilikni yo'qotish uchun uning ichiga guruxlashgan panel **Croup Boxes** qo'yish va uni o'lchamini **Pait Box** komponenta o'lchami bilan bir xil qilish lozim. Ana 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.

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

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.

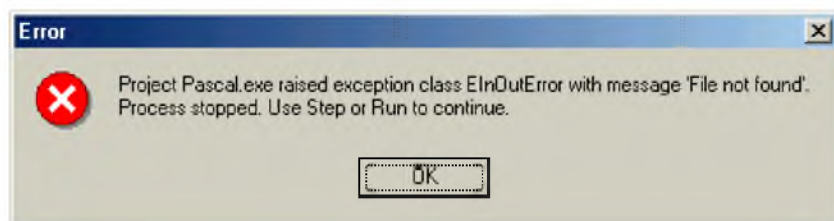
Odatda dastur tuzilayotganda ba'zi kamchilik yoki xatoliklarga yo'l qo'yamiz, dasturni ishga tushirgan vaqtimizda esa bu xatoliklar to'g'risida axborot beruvchi oyna hosil bo'ladi va bu xatoliklar quyidagilarga bo'linadi: Sintaktik; Dastur bajarilish vaqtidagi xatoliklar; Algoritmik.

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. Xatoliklar oynasi quyidagicha ko'rinishga ega bo'lishi mumkin:



3.2-rasm. Dastur bajarilish vaqtidagi xatolik.

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.

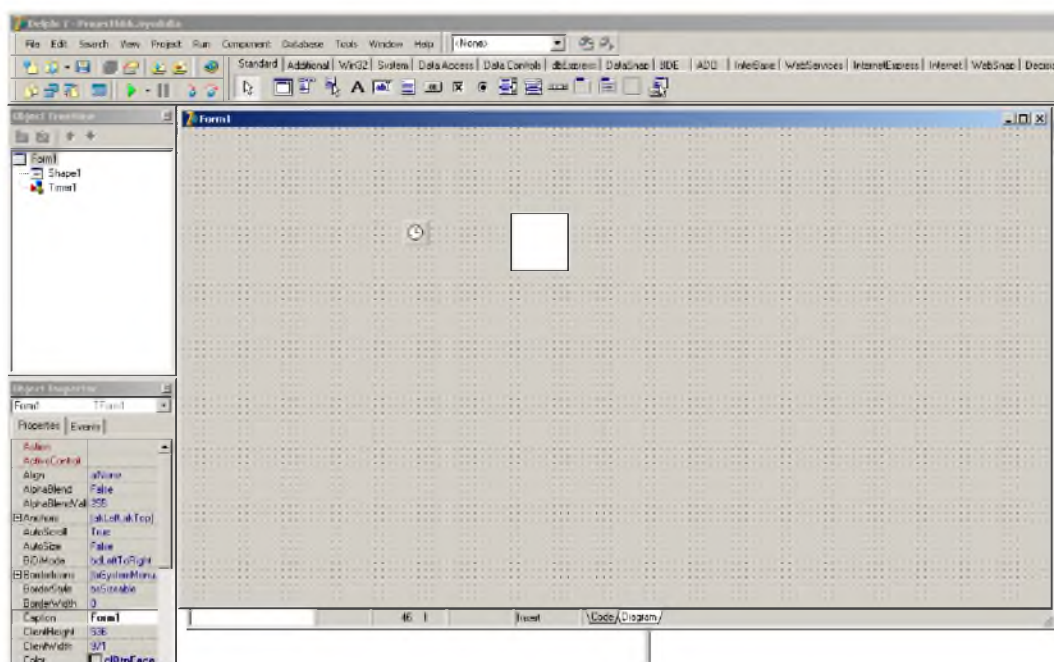
Dasturni qadamlab bajartirish uchun birinchi navbatda **Run** menyusidagi **Add watch (Ctrl+F5)** buyrug'i tanlanadi va o'zgaruvchi qiymatlarini o'zida saqlaydigan (**Watch List**) oyna xosil bo'ladi. Unga xatolikni tekshirish jarayonida zarur bo'lgan o'zgaruvchilar kiritiladi. So'ngra kursor noto'g'ri 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.

Bosh ishchi forma Form1 formasining xususiyatlarini o'zgartirish va formaga kerakli komponentlarni (matnlarni kiritish va chiqarish maydoni, buyruqlar tugmasi) qo'yish yo'li bilan yaratiladi.

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. 1 ta Shape komponentasidan foydalanamiz. Natijada ekranda dastur loyihasi hosil bo'ladi:



So'ngra Form da sichqoncha tugmasini ikki marta bosamiz (Click) hodisasining kodlar oynasiga quyidagi dasturni kiritamiz:

unit Unit666.Ziyodulla;

interface

uses

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

Dialogs, ExtCtrls;

type

TForm1 = class(TForm)

Timer1: TTimer;

Shape1: TShape;

procedure FormCreate(Sender: TObject);

procedure Timer1Timer(Sender: TObject);

procedure Shape1ContextPopup(Sender: TObject; MousePos: TPoint;
var Handled: Boolean);

private

{ Private declarations }

public

{ Public declarations }

end;

var

Form1: TForm1;

i:integer;

implementation

{ \$R *.dfm }

procedure TForm1.FormCreate(Sender: TObject);

begin

i:=0;

timer1.Enabled:=true;

end;

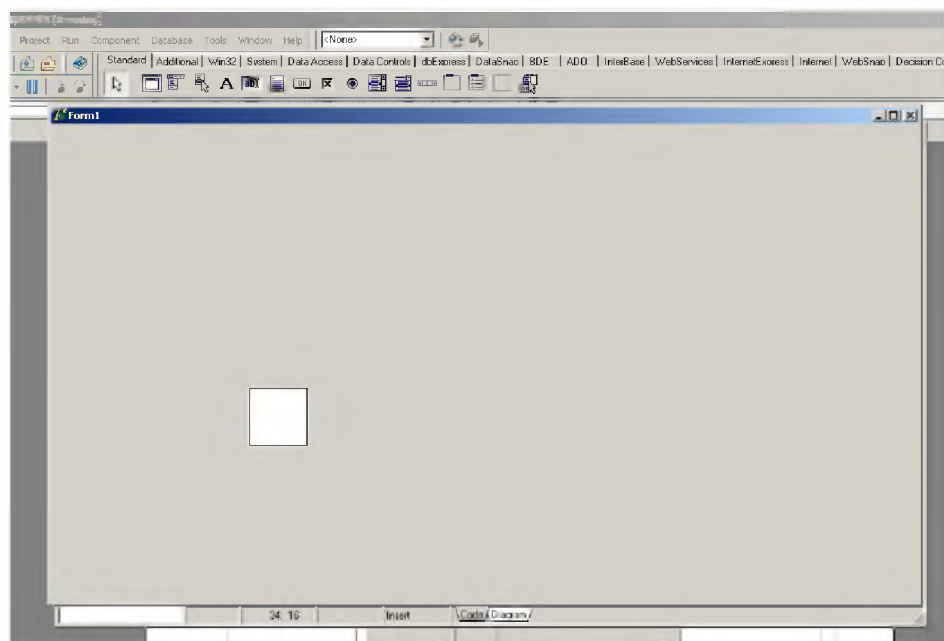
procedure TForm1.Timer1Timer(Sender: TObject);

begin

```
randomize;
Shape1.Left:=random(Form1.ClientWidth)-10;
Shape1.Top:=random(Form1.ClientHeight)-10;
end;
procedure TForm1.Shape1ContextPopup(Sender: TObject; MousePos: TPoint;
var Handled: Boolean);

begin
i:=i+3;
Form1.Caption:=inttostr(i);
end;
end.
```

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



1.7. Dasturni to'ldirish bo'yicha tavsiyalar.

Delphi dasturlash tilida nafaqat amaliy masalalarni echish mumkin, balki yana bir qancha imkoniyatlaridan foydalanish mumkin. Delphi dasturlash tilida mul'timedia imkoniyatlaridan foydalanish (Animate komponentasi, MediaPlayer komponentasi, ovozlar yozish, videorolik va animatsiyalarni ko'rish, animatsiyalar yaratish), ma'lumolar ba'zasi bilan ishlash. Delphining grafik imkoniyatlaridan foydalangan xolda suratlarni ekranga chiqarish, mul'tiplikatsiya, bitli tasvirlardan foydalanish, "mul'film" ko'rish kabi imkoniyatlaridan xam foydalanish mumkin.

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

Power Point dasturi xaqida tushuncha

Microsoft PowerPoint 97 — universal, imkoniyatlari keng bo'lgan, ko'rgazmali grafika amaliy dasturlari sirasiga kiradi va matn, rasm, chizma, grafiklar, animatsiya effektlari, ovoz, videorolik va boshqalardan tashkil topgan slaydlarni yaratish imkonini beradi.

Slayd — ma'lum bir ulchamga ega bo'lgan muloqot varaqlari xisoblanadi. Unda biror maqsad bilan yaratilayotgan namoyish elementlari joylanadi.

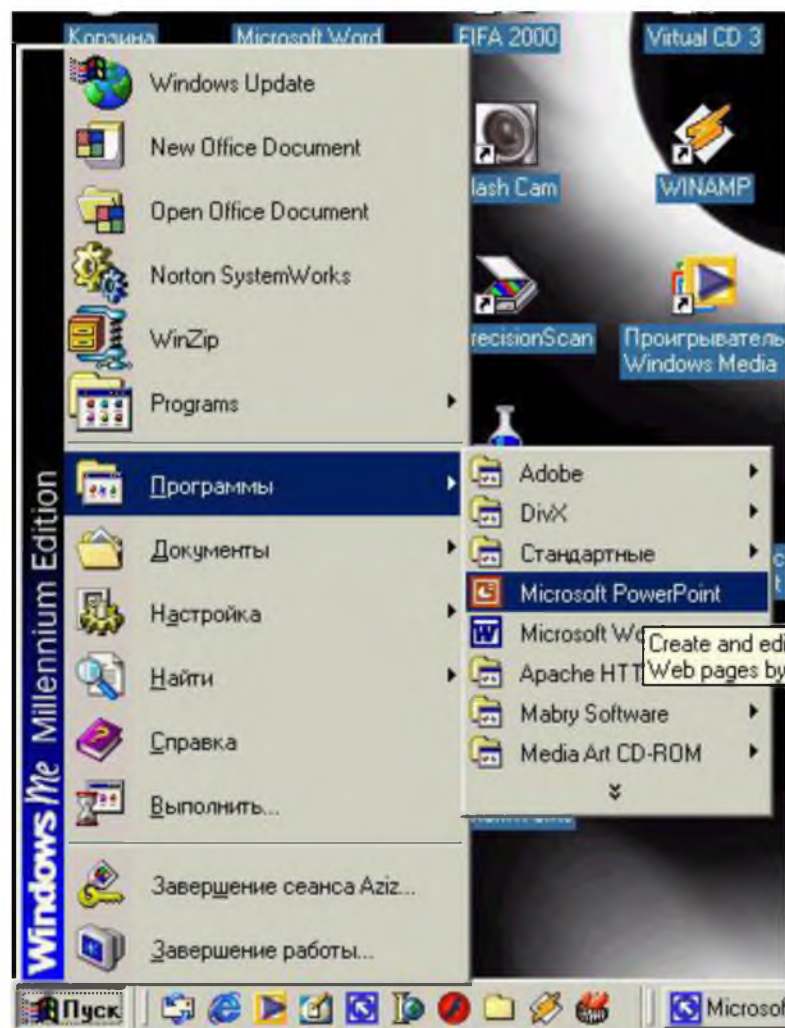
Slaydlar ketma-ketligidan iborat tayyor ko'rgazmani kompyuter ekranida, videomonitorida, katta ekranda namoyish qilish mumkin. Ko'rgazmani tashkil qilish — slaydlar ketma-ketligini loyixalash va jixozlash demakdir.

Taqdim etish axborot texnologiyasining samaradorligi ko'p jixatdan taqdim etuvchi shaxsga, uning umumiy madaniyati, nutk madaniyati va x.k.larga bog'lik ekanligini ham unutmaslik lozim.

PowerPoint dasturi MICROSOFT firmasining WINDOWS qobig'i ostida yaratilgan bo'lib, ushbu dastur prezentatsiyalar (taqdimot qilish, ya'ni tanishtirish) bilan ishlash uchun eng qulay bo'lgan dasturiy vositalardan biridir. Bu dastur orqali barcha ko'rgazmali qurollarni yaratish va ba'zi joylarda esa ma'lumotlar bazasi sifatida ham qo'llash mumkin. Ayrim xollarda bu dasturdan multimedia vositalarini boshqarish va ularni qo'llab, namoyish etuvchi qurilmalarga yuborish vazifalarini ham bajarish mumkin. Dasturdagi asosiy tushunchalar bu — **slayd** va **prezentatsiya** tushunchalaridir.

PowerPoint dasturini ishga tushirish. Bu dasturni ishga tushirishni WINDOWS ish stolidan boshlash zarur. Ish stolidagi quyidagi buyruqlarni bajarish orqali dastur ishga tushiriladi:

«Пуск» — «Программы» — «Microsoft PowerPoint»



Eslatma: yukoridagi buyruklarni bajarish sichkonchanning chap tugmasini bosish orqali amalga oshiriladi.

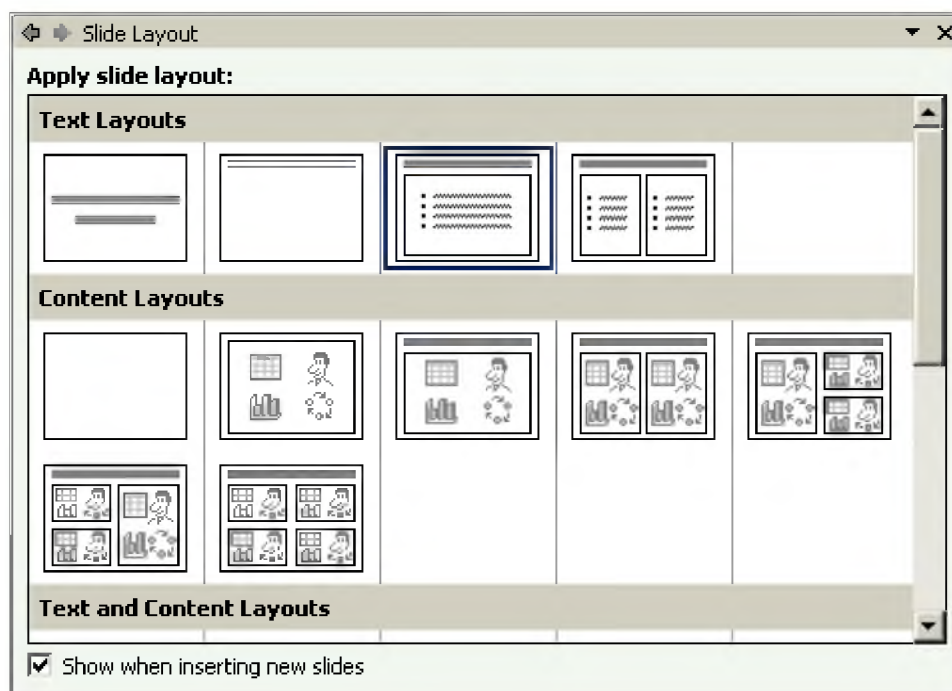
PowerPoint ishga tushirilganda ekranda ko'rsatilgan taqdimot yaratish muloqot darchasini ko'rish mumkin.

Bu muloqot darchasi orqali taqdimot yaratish turini tanlab olish mumkin bo'ladi. Quyidagi 4 ta asosiy bo'lim berilgan, ularning xar biri hakida qisqacha to'xtalib o'tamiz.

- **«Мастер автосодержания»** (Avtomundarija ustasi) — ma'ruzachi ismi, familiyasi, mavzu nomi, taqdimot turi, uslubi, rangli chizmasini kiritib, bir necha slayddan iborat bo'lgan tayyor taqdimotni chiqaradi.

- **«Шаблон презентации»** (Taqqimot qoliplari) — taqqimotni jixozlash turini tanlash va ularda ish olib borish imkonini beradi.
- **«Пустую презентацию»** (Bo'sh taqqimotni) — mustaqil ravishda taqqimot tuzish imkonini beradi.
- **«Открыть презентацию»** (Taqqimotlarni ochish) — kompyuter xotirasida va diskda mavjud bo'lgan ko'rgazmali fayllarni ochadi.

Har bir foydalanuvchi dasturda ish olib borishi uchun yuqoridagi bo'limlardan birini o'z maqsadiga ko'ra tanlab oladi. Dastur ishini bo'sh taqqimotdan ham boshlash mumkin. Buning uchun sichqoncha ko'rsatkichi **«Пустую презентацию»** (Bo'sh taqqimotni) bo'limiga olib kelinadi va **«OK»** tugmasi bosiladi. Bunda quyidagi muloqot darchasi hosil bo'ladi.



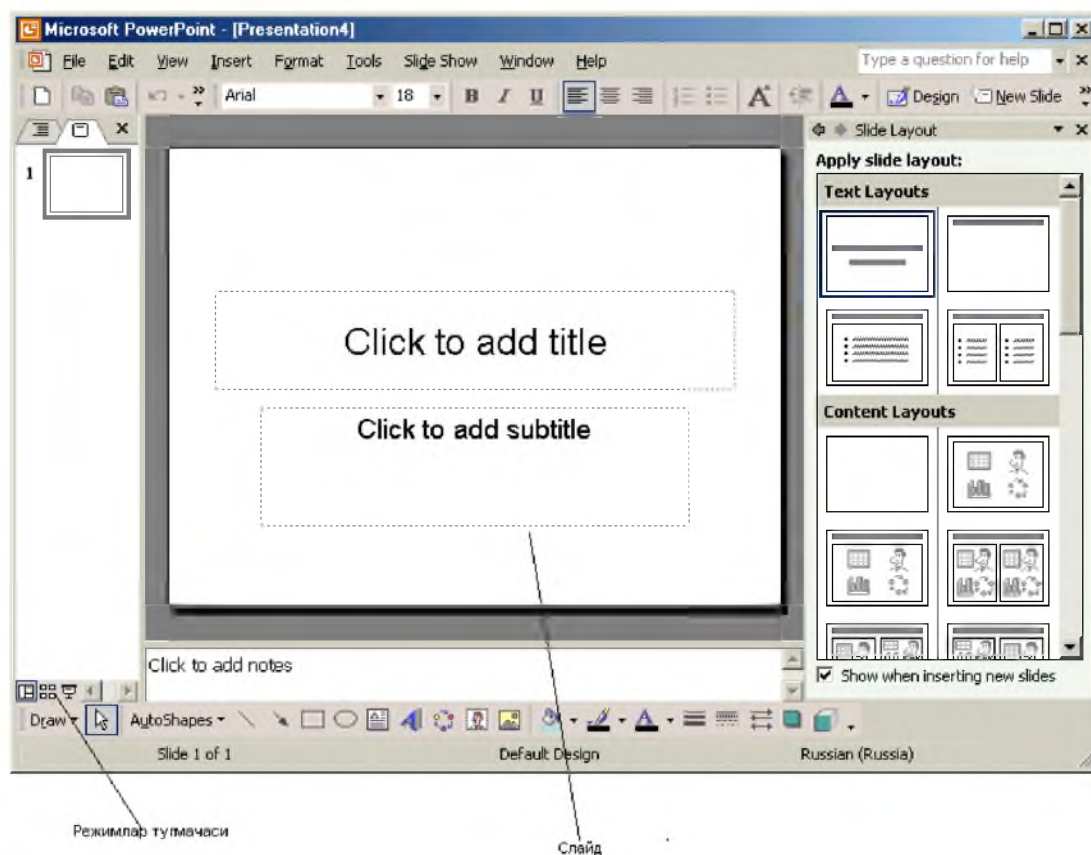
Bu yerda ko'rsatilgan xar bir slaydni tanlab olish imkoniyati bor. Buning uchun yo'nalish tugmalari yoki sichqonchadan foydalanish mumkin.

Muloqot darchasidan kerakli xolat, masalan: **«Пустой слайд»** (Bo'sh slayd) ko'rinishi tanlanadi va **«OK»** tugmachasi bosiladi.

Bu amallar bajarilgandan so'ng Power Point dasturining asosiy oynasi xosil bo'ladi. U sarlavxalar qatori, gorizontaal menyu, uskunalar paneli («standart», «bichimlash» va rasmlar bilan ishlash), rejim (xolat) tugmachalari (slaydlar rejimi,

strukturalar rejimi, saralash rejimi, namoyishlar rejimi) va ishchi maydonni o'z ichiga oladi.

PowerPoint ekranining pastki chap burchagida ish rejimi va uning turlarini boshqaruvchi tugmachalar (piktogrammalar) guruxi joylashgan.



PowerPointdan chiqish uchun «**Fayl**» menyusidan « **Выход** » («Chikish») buyrug'i tanlanadi.

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'zlashtirdim;
2. Delphi dasturlash tilida Timer bilan ishlash va Timerdan foydalangan xolda o'yin dasturlarini tuzishni o'rgandim;
3. Delphi da o'zimga berilgan kurs ishini bajarish bilan bir qatorda, Delphi ni qo'shimcha imkoniyatlaridan xam foydalandim;
4. Qo'shimcha topshiriqni bajarish mobaynida Power Point dasturini yanada mustaxkamladim.

Kurs ishini bajarish davomida bir qancha natijalarga erishish bilan birgalikda, bir qator sintaktik 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. Н.Отаханов. “Delphi”да дастурлаш. 2005 й.
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. N.Qurbonov “Ob'ektga mo'ljalangan dasturlash” fanidan ma'ruzalar matni. 2007 y
6. «Elektrotexnika materiallari» fanidan ma'ruzalar matni
7. 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 Unit666.Ziyodulla;

interface

uses

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

type

TForm1 = class(TForm)
Timer1: TTimer;
Shape1: TShape;
procedure FormCreate(Sender: TObject);
procedure Timer1Timer(Sender: TObject);
procedure Shape1ContextPopup(Sender: TObject; MousePos: TPoint;
var Handled: Boolean);
private
    { Private declarations }
public
    { Public declarations }
end;

var
    Form1: TForm1;
    i:integer;

implementation
{$R *.dfm}
procedure TForm1.FormCreate(Sender: TObject);
begin
i:=0;
timer1.Enabled:=true;
end;
```

```
procedure TForm1.Timer1Timer(Sender: TObject);
begin
    randomize;
    Shape1.Left:=random(Form1.ClientWidth)-10;
    Shape1.Top:=random(Form1.ClientHeight)-10;
end;

procedure TForm1.Shape1ContextPopup(Sender: TObject; MousePos:
TPoint;
var Handled: Boolean);

begin
i:=i+3;
    Form1.Caption:=inttostr(i);
end;
end.
```