

O'ZBEKISTON RESPUBLIKASI
OLIY VA O'RTA MAXSUS TA'LIM VAZIRLIGI

Surxondaryo viloyati Hokimliginig
O'rta maxsus kasb-hunar ta'limi boshqarmasi

TERMIZ DAVLAT UNIVERSITETI

O'rta maxsus kasb-hunar ta'limi tizimi kadrlar malakasini
oshirish tizimi va ularni qayta tayyorlash fakulteti

**DENOV XAZARBOG' QISHLIQ XO'JALIGI KASB-HUNAR
KOLLEJI INFORMATIKA FANI O'QITUVCHISI
SHERNAZAROV PIRMAMATning "DELPHI DASTURLASH
TILI VA UNDA AMALIY DASTURLAR YARATISH"
mavzusidagi**

MALAKAVIY ISH

ILMIY RAHBAR:

prof. CH. NORMURODOV

Termiz - 2011

REJA

Kirish

I. Delphi dasturlash tili xaqida tushuncha

1.1 Delphi ning bosh darchasi

1.2 Formada klavitura tugmalari vazifalari

1.3 Satrlar bilan ishlovchi funksiyalar

II. Delphi dasturlash tilida komponentalar

2.1 Boshqaruvchi tugmalari Button va BitBtn, SpeedButton tugmasi

2.2 Ko'p satrli taxirlash darchalari-Memo va RichEdit komponentalari, Formalar

2.3 Matematik funksiya va protseduralar

III. Asosiy qism

3.1 Test dasturini yaratish algoritmi

3.2 Dasturda ishlatiladigan komponentalar

3.3 Test dasturining komponentalarida kodlar

Xulosa

Foydalanilgan adabiyotlar

Ilova

KIRISH

Yangi asr o'z navbatida jamiyatimiz oldiga muhim masalalarni hal qilish vazifasini yuklamoqda. Ishlar majmuini hal qilish har bir fuqaroning bu jarayonda shaxsiy ishtirokida, ijodiy ma'suliyatli munosobatiga hamda manfaatdorligiga bevosita bog'liq.

Ushbu bitiruv malakaviy ishda dasturlash tilining, ya'ni Pascal tilining rivoji bo'lgan Object pascal tilida dastur tuzilgan. Bu bitiruv malakaviy ishda Delphi tili yordamida "Test" dasturi tuzilgan. "Test" uncha murakkab bo'lmagan ko'rinishga ega, ya'ni 7 ta savoldan iborat bo'lgan test mavjud.

Ushbu bitiruv malakaviy ish 3 ta bob, kirish, xulosa va foydalanilgan adabiyotlardan iborat.

1 chi bobda Delphi dasturlash tili xaqida tushuncha berilgan, unda Delphi ning bosh darchasi, Formada klavitura tugmalari vazifalari, Satrlar bilan ishlovchi funksiyalar xaqida berilgan

2 chi bobda Delphi dasturlash tilida komponentalar xaqida tushuncha berilgan, unda Boshqaruvchi tugmalari Button va BitBtn, SpeedButton tugmasi, Ko'p satrli taxrirlash darchalari-Memo va RichEdit kompanentalari, Formalar, Matematik funksiya va protseduralar xaqida berilgan.

3 chi bobda esa Asosiy qism xaqida, ya'ni test dasturi xaqida tushuncha berilgan, unda Test dasturini yaratish algoritmi, Dasturda ishlatiladigan komponentalar, Test dasturining komponentalarida kodlar xaqida berilgan.

Malakaviy ish maqsad va vazifalari. Ushbu malakaviy ish ishning maqsadi deb quyidagilarni belgiladik: Delphi dastirlash tili imkoniyatlarini o'rganish va unda amaliy dasturlar to'plamini yaratishdan iborat

Malakaviy ishning amaliy ahamiyati. O'quvchilarni dars mobaynida dasturlash tili qobiq dasturlari bilan tanistirish va ularda dastur yozish ko'nikmalarini hosil qilish.

Malakaviy ishning strukturasi. Ushbu malakaviy ish kompyuteda terilgan 25 betdan 8 ta rasmdan iborat bo'lib. I.1 qismda Animatsiyalarni yaratish texnolo'giyasi

Adabiyotlar tahlili.

MainMenu – dastur bosh menyusi. Komponenta murakkab ierarxik strukturali menyu yaratish uchun xizmat qiladi.

PopurMenu – yordamchi yoki local menyusi. Bu menyu oynasi sichqoncha o'ng tugmasini bosish bilan chiqadi.

Label – metka (belgi). Bukomponenta forma oynasiga uncha uzun bo'lmagan bir qatorli yozuvni chiqarishda ishlatiladi va uning piktogrammasi panelda “A” ko'rinishda berilgan.

Edit - kiritish qatori. Forma oynasida matnli qator kiritish va taxrirlashda ishlatiladi.

Memo - qatorli matn muxariri. qatorli matnlarni kiritish yoki chiqarishda ishlatiladi.

Button - Buyruq tugmasi. Bu komponenta dasturchi tomonidan berilga bir necha buyruqlarni bajarishda ishlatiladi.

Checkbox - bog'liq bo'lmagan o'chirib yoqadigan tugmacha . Bu koponetnta yordamida programmaning mantiqiy xossasi o'zgartiriladira.

RadioButton - bog'liq bo'lgan ochirb yoqadigan tugmacha . Oldin tanlangan pereklyuchatellarni avtomatik ravishda ozod etadi.

I.1 Delphi ning bosh darchasi.

Forma komponentalari bu dasturni boshqarish uchun maxsus tugmachalar bo'lib uni formaga joylashtirishdan oldin bosh oynadan kerakli komponentalar politrasi tanlanadi. Masalan, Standart komponentalar politrasi quyidagi tugmachalar majmuasi mavjud:

MainMenu – dastur bosh menyusi. Komponenta murakkab ierarxik strukturali menyu yaratish uchun xizmat qiladi.

PopurMenu – yordamchi yoki local menyusi. Bu menyu oynasi sichqoncha o'ng tugmasini bosish bilan chiqadi.

Label – metka (belgi). Bu komponenta forma oynasiga uncha uzun bo'lmagan bir qatorli yozuvni chiqarishda ishlatiladi va uning piktogrammasi panelda "A" ko'rinishda berilgan.

Edit - kiritish qatori. Forma oynasida matnli qator kiritish va taxrirlashda ishlatiladi.

Memo - qatorli matn muxariri. qatorli matnlarni kiritish yoki chiqarishda ishlatiladi.

Button - Buyruq tugmasi. Bu komponenta dasturchi tomonidan berilga bir necha buyruqlarni bajarishda ishlatiladi.

Checkbox - bog'liq bo'lmagan o'chirib yoqadigan tugmacha . Bu komponentda yordamida programmaning mantiqiy xossasi o'zgartiriladira.

RadioButton - bog'liq bo'lgan ochirib yoqadigan tugmacha . Oldin tanlangan pereklyuchatellarni avtomatik ravishda ozod etadi.

Listbox - ro'yxatdan tanlash. Ro'yxat variantlarini taqdim etadi va tanlash imkonini yaratadi.

ComboBox - kobinirovanniy ro'yxatdan tanlash. Ro'yxatdan kombinasiya qilib tanlash.

ScrollBar - polosali bishqarish. Vendoys oynasi chetlarida gorizantal yoki vertikal polosa tashkil etadi.

GroupBox - elementlar guruhi. Ma'no bo'yicha bir necha bog'liq komponentalarini guruhlashda ishlatiladi.

RadioGroup – bog'liq guruh pereclyuchatellari. Bir necha bog'liq pereklyuchatellar xossalarini saqlaydi.

Panel – panel. Bu komponenta, xuddi GroupBoxga o'xshab bir necha komponentalarni birlashtirish uchun xizmat qiladi.

Actionlist – ta'sir qilish ro'yxatlari. Foydalanuvchi dasturga markazlashgan xolda ta'sir qilishi uchun ishlatiladi.

Tugmalar	Tasnifi va vazifalari
F1, Ctrl+F1	Yordam
F3	Izlashni davom rtish
F4	Programmani kursor turgan joygacha bajarish
F5	<i>Break Point</i> o'rnatish
F7	Proteduraga kirishlik bilan tekshirish
F8	Proteduraga kirmasdan bilan tekshirish
F7	Programmani ishga tushurish
F10, Ctrl+F10	Asosiy menyuni joriy qilish
F11	<i>Objekt Inspektor</i> dan kodlar darjasiga yoki formaga o'tish
F12	Formadan kodlar darchasiga o'tish, yoki aksincha
CTRL+F2	Programma bajarilishini yakunlash(buning uchun kursor asosiy menyular darchasida yoki kodlar darchasida bo'lishi kerak)
CTRL+F3	Stekni ko'rish
CTRL+F4	Joriy modulni yopish

CTRL+F5	WatchList hatoliklar ro'yhatini chiqarish
CTRL+F7	Evaluate/Modify darchasini ochish(o'zgaruvchilar qiymatini ko'rish va o'zgartirish)
CTRL+F9	Programmani kompilatsiya qilish
CTRL+F11	Proektni ochish
CTRL+F12	Proekt modullari ro'yhati
CTRL+probel	Obyektlar ro'yhatini chiqarish
Shift-F10	Kontekst menyu
Shift-F11	Proektga modul qoshish
Shift-F12	Proekt formalari ro'yhati
ALT+F4	Proektni va barja darchalarni yopish
Alt-0	Darchalar ro'yhatini chiqarish
Ctrl-Enter	Kursor turgan so'z nomi bilan kursor ochish
CTRL+sichqoncha orqali biror so'zni tanlash	Tanlangan so'zni izohlash
ALT+matnni belgilash (функции выделения текста)	Vertikal blokni belgilash
Ctrl+Shift+↓/↑	Prosedurani e'lon qilish bo'limidan uning tanasiga o'tish va aksincha
Ctrl-Shift-C	Qism dasturni to'ldirish.(Agar e'lon qilingan bo'lsa, tanasini hosil qilish va aksincha)
Ctrl-Shift-E	Exploring darchasini ochish
Ctrl-Shift-R	Makroni yozishni boshlash va tugatish
Ctrl-Shift-P	Makroni bajarish
Ctrl-Shift-U	Belgilangan blokni abzatsini kamaytirish
Ctrl-Shif-I	Belgilangan blokni abzatsini ko'paytirish
Ctrl-Shift-G	GUI o'rnatish

Ctrl-Shift-B	Klasning shajarasini ko'rish
Ctrl-Alt-W	Watch List darchasini ochish
Ctrl-Alt-T	Proekt oqimlar ro'yhati
Ctrl-Alt-S	Stek darchasini ochish
Ctrl-Alt-L	Lokal o'zgaruvchilar darchasini ochish
Ctrl-Alt-V	Hodisalar ro'yhati
Ctrl-Alt-B	Brekt Points lar ro'yhati
Ctrl-Alt-M	Modullar
Ctrl-N	Bosh satr qo'shish, kursor joriy satrda qoladi
Ctrl-M,,Enter	Bosh satr qo'shish, kursor keyingi satrda o'tadi
Ctrl-E	Tez qidirish. Ctrl+E dan keyin kerakli so'z kiritiladi
Ctrl-R	Izlash va almashtirish
Ctrl-T	Kursor turgan joydan so'z oxirigacha o'chirish
Ctrl-Y	Satrn o'chirish
Ctrl-S	Joriy faylni saqlash
Ctrl-F	Izlash darchasini ochish
Ctrl-J	Shablonlar ro'xatini ochish
Ctrl-Z	Oxirgi amalni bekor qilish (отмена)
Ctrl-X	Kochirish
Ctrl-C	Nusha olish
Ctrl-V	Qoyish
Ctrl-B	Buferlar ro'yhati
CTRL+K+E	Kursor turgan so'zni katta harf bilan yozish
CTRL+K+E	Kursor turgan so'zni kichik harf bilan

	yozish
CTRL+K+N	Belgilangan so'zni katta harf bilan yozish
CTRL+K+O	Belgilangan so'zni kichik harf bilan yozish
Ctrl-Shift-0..9	0 ...9 gacha metka qo'yish
Ctrl-0..9	0 ...9 gacha metkaga o'tish

1.1 Formada klavitura tugmalari vazifalari.

Tab	Keyingi komponentani tanlash
Shift+Tab	Oldingi komponentani tanlash
↑/←/→/↓ steklari	Eng yaqin turgan komponentani tanlash
Ctrl+ strelkalar	Belgilangan komponentani bir piksil siljitish
Shift+↑/←/→/↓	Belgilangan komponentani bir piksil siljitish Kattalashtirish yoki kichiklashtirish
Ctrl+Shift+↑/←/→/↓	Belgilangan komponentani bir necha piksil siljitish
Delete	Belgilangan komponentani ochirish
Esc	Formani yoki Croupbox ni tanlash
Alt-Ctrl-F11	Proekt menenjeriga o'tish

1.3 Satrlar bilan ishlovchi funksiyalar

1. **InToStr (Value: Integer): string;** funksiyasi butun toifada o'zgaruvchilarni satrga o'girish uchun ishlatiladi;

2. **StrToInt (const S: string): Integer;** funksiyasi satrli (**string**) o'zgaruvchilarni butun songa o'zgarish uchun ishlatiladi;

3. **FloatToStr (Value: Extended); string;** funksiyasi haqiqiy toifadagi o'zgaruvchini satrli o'zgaruvchiga o'girish uchun ishlatiladi;

4. **StrToFloat (const S:string): Extended;** funksiyasi satrli (**string**) o'zgaruvchilarni haqiqiy songa o'zgarish uchun ishlatiladi.

II. Delphi dasturlash tilida komponentalar.

2.1 Boshqaruvchi tugmalari **Button** va **BitBtn**, **SpeedButton** tugmasi.

Sodda va eng ko'p ishlatiladigan tugmalar **Button** va **BitBtn** lardir. Bu komponentalarning ko'p xususiyat, xodisa va metodlar bir xil. Asosiy farqlardan biri esa, **BitBtn** komponentasida rasm qo'yish mumkinligidadir.

Tugmalarning asosiy xususiyatlaridan biri **Caption** (sarlavha). **Caption** xususiyati ma'lum harfdan oldin qo'yilgan **<&>** belgisi orqali tugmaga tezda murojat qilish mumkin.. Misol uchun **Caption** xususiyatida **<& Chiqish>** yozilgan bo'lsin. Bu Formada **<chiqish >** shaklida ko'rinadi. Bu tugmaga murojat qilish uchun **Alt-C** tugmalarini bosishlik kifoya.

Xar qanday tugmaning asosiy hodisasi **OnClick** bo'lib, bu hodisa tugma bosilganda sodir bo'ladi. Tugma bosilganda nima ish bajarilishi kerakligi aynan shu hodisada keltiriladi. Bundan tashqari sichqoncha va klaviatura orqali bo'ladigan bir qancha hodisalar mavjud. Bularni keyingi mavzularda o'rgani chiqamiz. Agar tugmani **Action** xususiyati bilan bo'lmagan bo'lsangiz **OnClick** xodisasida nima vazifa bajarilishi kerakligini yozish lozim.

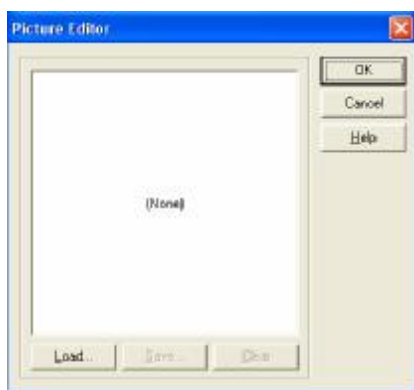
Cancel xususiyatida **true** qiymat o'rnatilgan bo'lsa, foydalanuvchi **Esc** tugmasini bosishi, tugmani bosishi bilan ekvivalent ishlaydi. Yani tugmani **OnClick** hodisasi bajariladi. Bu xodisani, turli dialog darchalarida, dialogni bekor qilish tugmalari ishlatish mumkin. Foydalanuvchi dialogni tugatish uchun **Esc** tugmasini bosishi mumkin.

Default xususiyatida **true** qiymat o'rnatilgan bo'lsa, foydalanuvchi **Enter** tugmasini bosishi bilan ekvivalent ishlaydi. Agar bir nechta tugmada **Default** xususiyatida **true** qiymat o'rnatilgan bo'lsa, **TabOrder** xususiyatidagi tartib bo'yicha bajariladi.

Tugma programma kodi orqali ham murojat qilish mumkin. Formaning istalgan qismida <Ha> yoki <h> tugmalarini bosishlik bilan hissoqlash jarayonini amalgam oshirmoqchisiz. Buning uchun formaning **KeyPreview** hususiyati **true** qilinadi. Formaning **OnKeyPress** hodisasiga quyidagilar yoziladi.

If (key='C') or (key='c') then Button1. Click;

Agar foydalanuvchu <Ha> yoki <h> tugmalaridan birini bossa, button1 tugmasi bosilgandagi hodisa bajariladi.



Yuqorida aytilgan xam **Button** uchun, ham **BitBth** uchun o'rinli. Endi **BitBth** ga hos bo'lgan hususiyatlar bilan tanishib chiqamiz. Bu tugmaga rasm **Glyph** hususiyati orqali belgilanadi. Obyektlar inspektorida **Glyph** <<...>> ni bosish orqali quyidagi darcha ochiladi.

Load tugmasini bosish orqali bmp turidagi rasmni tanlash mumkin. Delphi da tugmalar uchun bir qator rasmlar keltirilgan. Bu rasmlar / *Program Files/ Common Files/Borland Shared/Images/Buttons/* katalogida saqlanadi. Rasman tanlaganingizdan so'ng Ok tugmasini bosing. Tanlagan rasm **BitBth** da sarlovhadan chapga joylashadi.

Rasm va sarlovhaning tugmada joylashishi **Margin, Layout, Spadeg** hususiyatlari orqali beriladi. Agar **Margin** =-1 bo'lsa, rasm va sarlovha tugma markazida bo'ladi. Bu holatda rasmni sarlovhaga nisbatan qayerda turishi **Layout** hususiyati orqali aniqlanadi. **BiGlyphTop** (tepada), **BiGlyphButtom** (pastdan) **BiGlyphLeft** (chapda) **BiGlyphRight** (o'ngda).

Agar **Margin** > 0 bo'lsa, Layout qiymatiga tugmaning qarab u yoki bu chegarasidan **Margin** da berilgan pikselcha joy tashlanadi.

Spacing xususiyati rasm va sarlovha orasidagi piksellarni bildiradi. Odatda **Spacing** =4 qiymat o'rnatilgan bo'ladi. Agar **Spacing** =0 bo'lsa, rasm

va bir-biriga tegib turadi. Agar **Spacing** =-1 bo'lsa sarlovha rasm va tugma chegaralari o'rtasida bo'ladi.

Kind xususiyati tugma turini aniqlaydi. Odatda bu xususiyatga **bkCustom** o'rnatilgan bo'ladi. Bundan tashqari **bkOk**, **bkCancel**, **bkYes**, **bkNo**, **bkClose**, **bkAbort**, **bkIgnore**, **bkAll** ni o'rnatish mumkin.

SpeedButton tugmasi

SpeedButton tugmasidan oddiy boshqaruv tugmalar kabi foydalanish mumkin. Bundan tashqari **SpeedButton** tugmasidan fiksirlangan tugma sifatida foydalanish mumkin. Bu tugmalar odatda vazifalar panelida, menyular qatoridagi ko'po ishlatiladigan buyuruqlar nusxasini ko'rsatish uchun ishlatiladi.

SpeedButton tugmasida boshqa tugmalar kabi **Caption** xususiyatlari mavjud. Lekin bu odatda bo'sh bo'ladi, uning o'rniga rasm (piktogramma) ishlatiladi. Tugmaga rasm **Glyph** xususiyati orqali beriladi **NumGlyph**, **Layout**, **Margin**, **Spacing** xususiyatlari **SpeedButton** tugmasi uchun ham o'rinli.

SpeedButton tugmasining sosiy xususiyatlaridan biri **GroupIndex** (guruhlar indeksi). Agar **GroupIndex**=0 bo'lsa, tugma huddi **Button**, **Bitbtr** tugmalari kabi qachonki qo'yib yuborsa, tugma o'z holiga qaytadi.

Agar **GroupIndex**>0 va **AllowAllUp**=true bo'lsa, foydalanuvchi tomonidan tugma bosilganda, tugma bosilib qoladi. Qachonki tugma ikkinchi marotaba bosilganda o'z holiga keladi. (ikkinchi marotaba bosganda tugma o'z holiga kelishi uchun **AllowAllUp**=true bo'lishi kerak). **Down** xususiyati qaysi tugma bosilganini bildiradi. Yani qaysi tugma bosilsa, shu tugmaning **Down**=true ga o'zgaradi.

Programmani ishlab chiqish jarayonida **Down**=true qilingan bo'lsa programma ishga tushganda tugma bosilgan holda bo'ladi.

SpeedButton tugmasining **Flat** hususiyatini **true** ga o'zgartirish orqali chiroyli interfeys hosil qilish mumkin. Kursor tugma ustidan tushganda tugma o'z holiga qaytadi.

Barcha boshqariluvchi komponentalarning **Hint** hususiyati orqali ko'rsatma satrni berish mumkin. Ko'rsatma satri kursor komponenta ustiga kelganda, bu komponenta nima ish bajarishi haqida ma'lumot beradi. Ko'rsatma satri formada ko'rinishi uchun komponentaning **ShowHint** hususiyati **True** qilinadi.

2.2 Ko'p satrli taxrirlash darchalari-Memo va RichEdit komponentalari, Formalar.

Memo va **RichEdit** komponentalari ko'p satrli matnlarni tahrirlash uchun ishlatiladi. Barcha taxrirlash darchalaridagi kabi **Memo** va **RichEdit** komponentalarida ham nusxa olish **Ctrl-C (copy)**, qirqib olish **Ctrl-X**, qo'yish **Ctrl-V, (paste)**, oxirgi amalni bekor (**Ctrl-Z**), qilish imkoniyatlarini keltirgan. **Memo** komponentasida format (Shrift, atributlar) barcha matn uchun bir xil bo'ladi va **Font** hususiyati orqali belgilanadi. Agar siz matnni faylga saqlasangiz, faqat simbollarni o'zida saqlovchi matnli fayl hosil bo'ladi. Bunda format saqlanmaydi. Saqlangan faylni **Memo** komponentasiga ochganda **Font** hususiyatida o'rnatilgan shirift bilan ochiladi. Saqlanishdagi shirift bilan emas. **RichEdit** komponentasi orqali RTF kengaytmali fayllar bilan ishlash mumkin. Shiriftni **SelAttributes** hususiyati orqali hohlagancha o'rnatish mumkin. Bu hususiyat **TTextAttributes** toifasida bo'lib, quyidagi ost hususiyatlarni o'z ichiga oladi: **Color** (rang), **Name** (shirift nomi), **Size** (o'lchami), **Style** (shakli) va boshqalar.

Aligment	Matnni tekislashni aniqlaydi taLeftJustify (o'ngga), taCenter (markazga), taRight Justify (chapga) qiymatlarni qabul qiladi.
-----------------	---

Numbering	Marker qo'yishni boshqarish. Quyidagicha qiymatlar qabul qiladi: nsNone -markerni bekor qilish, nsBullet -markerni o'rgatish.
LeftIndint	Chap tarafdin qoldiriladigan piksellar soni
RightIndert	O'ng tarafdin qoldiriladigan piksellar soni

TabCount va **Tab** hususiyatlari **Want Tabs** hususiyati **true** bo'lganida manoga ega. Agar **Want Tabs=false** bo'lsa, foydalanuvchi **Tab** tugmasini bosganida **Fokus** keyingi komponentaga uzatiladi. Biz Memo va RichEdit komponentalarining asosiy farqlarini qirqib chiqdik. Endi ularning umumiy hususiyatlariga to'xtalamiz.

Aligment va **WordWrap** hususiyatlari 3.1 bo'limida aytib o'tilgan vazifaning bajaradi. Yani matlarni tekslash va uzun satrlarni keyingi satrga o'tkazish uchun ishlatiladi. **ReanOnly** hususiyatini **True** qilish irqali matnni o'qish mumkin. **MaxLength** hususiyati kiritilishi mumkin bo'lgan matn uzunligini bildiradi. Boshqa hollarda kiritilishi mumkin bo'lgan belgilar sonini bildiradi. **ScrollBars** hususiyati orqali komponentaga siljitish yo'lakchasini o'mahs mumkin. **ScrollBars** hususiyati **ssNone** (siljitish yo'lakchasi o'rnatilmasin), **ssHorizontal** –(garizantal), **ssVertical** (vertikal), **ssBorth** (ham garizantal, ham vertikal) qiymatlarini qabul qilish mumkin.

Memo va **RichEdit** komponentalarining asosiy hususiyati **Lines**. **Lines** hususiyati **TStrings** toifasiga tegishli bo'lib matnni satrlar ro'yhati sifatida saqlaydi.

Matnni programma ishlab chiqarish jarayonida ham kiritish mumkin. Buning uchun **Object Inspektor** dan **Lines** hususiyati to'g'risidagi uch nuqta bosiladi.

Text hususiyati butun matnni o'zida saqlaydi. Matnning ma'lum satrida **Strings [Index:Inreger]** hususiyati orqali murojat qilish mumkin. **Index Delphi** da 0 dan boshlanadi. Demak, **RichEdit1.Lines.Stringes[0]** matnni birinchi satri. **Count** hususiyati mantdagi satrlar sonini aniqlash uchun

ishlatiladi. Taxrirlash darchasini tozalash uchun **Clear** protsedurasiga murojat qilinadi.

Matn oxiriga yangi satr qo'yish uchun, **Lines** hususiyatining **Add** yoki saqlash uchun esa **SaveToFile** metodlarig murojat qilinadi . Misol uchun ixtiyoriy tugmaga quyidagilarni kiritish mumkin.

Formalar

Ihtiyoriy ilovaning elementi forma (**konteyner**) hissoblanadi. **Formaga** boshqa ko'rinadigan va ko'rinmaydigan komponentalarning joylashtirish mumkin. **Forna** foydalanuvchi nuqtai nazardan u ilova bilan ishlayotgan darchani. Ilovaga kiritlgan har bir yangi forma o'zining moduli (**Unit**)ga ega. **Modulda** forma bajarishi kerak bo'lgan funksiya protseduralar kiritiladi. Odatda murakkab ilovalar bir necha formadan iborat bo'ladi. Yangi ilova (**programma**) tuzish uchun file menyusidan new oplication komandasi tanlanadi. Odatda birinchi forma asosiy forma hisoblanadi. Ilovaga yangi forma qo'shish uchun file menyusidan new form tanlanadi. Bir formadan boshqasiga o'tish uchun Show va ShowModal metodlaridan foydalanish mumkin. ShowModal metodi joriy formani yopgandan keyin boshqa formalar bilan ishlashga ruxsat beradi. Bu metod operativ xotiradan unumli foydalanish uchun ishlatiladi.

Show va ShowModal metodlarini ayni vaqtda ko'rinmaydigan formalar uchun ishlatiladi. Agar forma ko'rinish ko'rinmasligi noma'lum bo'lsa, quyidagicha programma kodi keltiriladi.

```
If (not Form2.visibli)then Form2. ShowModal
```

Show va ShowModal metodlari bajarilganda formaning OnShow hodisasi sodir bo'ladi. Hide metodi orqali formani ko'rinmaydigan qilish mumkin. Formani close metodi orqali yopish mumkin. Bir necha forma bilan ishlaganda bir formadan boshqasiga o'tish uchun shift+F12 tugmasi bosiladi.

Formaning ba'zi hususiyatlarini ko'rib chiqamiz.

1. +Border Icons

- Bi System Menu Formada sistema menyusining ko'rinish ko'rinmasligi ta'minlaydi.
- biMinimize Formani свернуть qilishga ruxsat berishni ta'minlaydi.
- biMinimize formani развернуть qilishga ruxsat berishni ta'minlaydi.

2. Window State

- wsMaximized -darchani to'liq ekran holida bo'lishini ta'minlash.
- wsMinimized- darchani yig'ilgan holida bo'lishini ta'minlash.
- wsNormal darchani normal holida bo'lishini ta'minlash.

Formaning ba'zi xususiyatlarini ko'rib chiqamiz.

1. Onactivate hodisasi qachonki forma aktiv bo'lganda yuz beradi.
2. Onclose hodisasi qachonki forma yopilganda yuz beradi.
3. Oncreate hodisasi qachonki forma yaratilganda yuz beradi.
4. OnResize hodisasi qachonki forma o'lchamlari o'zgartirilganda (kattalashtirilganida yoki kichiklashtirilganda) yuz beradi.
5. OnClose Query hodisasi qachonki forma yopilishi oldidan savol bilan murojaat qilish uchun ishlatiladi.

2.3 Matematik funksiya va protseduralar.

funksiya	Izoh	argumenlar
Abs(x)	absoliut qiymat	Butun yoki haqiqiy
Ceil(x)	Argumentga teng yoki argumentdan Katta bo'lgan butun sonni aniqlash	Extended
CompareValue(A,B[epsilon])	A va B ni Epsilon aniqlikda taqqolash	Butun yoki haqiqiy
DifMod(Dividen,Divisor,Resu)	Divedent-bo'linuvchi	Butun

It, Remainder)	Divisor-bo'luvchi Result-bo'linma Remainder-qoldiq	
EnsureRange(A Value, A Min, A Max)	A Valuega[A Min, A Max] oraliqdan eng yaqin butun sonni aniqlash	Butun yoki haqiqiy
Cos(x)	Cosinus	Burchak radianda beriladi
Exp(x)	Exponenta	haqiqiy
Floore(x)	Argumentga teng yoki argumentdan kichik bo'lgan sonni aniqlash	Extended
Frac(x)	Argumentning kasr qismi X-Int(x)	Extended
InRange(A Value, A Min, A Max)	A Value,[A Min, A Max] oralliqqa tegishliligini tejshirish	Butun yoki haqiqiy
Int(x)	Argumentning butun qismi	haqiqiy
Int Power(x,e)	X ning E darajasini aniqlash. X^E ; E-butun son	Extended va integer
Ldexp(x,p)	X ni 2^p ga ko'paytirish: $X * 2^p$	Extended va integer
Ln(x)	Natural logarifm Ln x	haqiqiy
LnXP1(x)	Natural logarifm Ln (x+1)	Extended
Log10(x)	O'nli lagarifm: lg x	Extended
Log2(x)	Logarifm ikki asosga ko'ra X: $\log_2 X$	Extended
LogN(n,x)	Logarifm N asosga ko'ra X:	Extended

	$\log_N X$	
Max(a,b)	Kattasini aniqlash	Integer, int64, Singl, double, Extended
Min(a,b)	Kichigini aniqlash	Integer, int64, Singl, double, Extended
Pi	3.1415926535897932385	-
Power(x,e)	X ning Edarajasini aniqlash X^E ;	Extended
Round(x)	Yalitlash	Extended
SameValue(a,b,Epsilon)	A va b ni epsilon aniqlikda taqqoslash	haqiqiy
Sqr(x)	Kvadrat: $X * X$	Extended
Sqrt(x)	Ildiz	Extended
Trunc(x)	Butun qismini aniqlash	Extended

III. Asosiy qism.

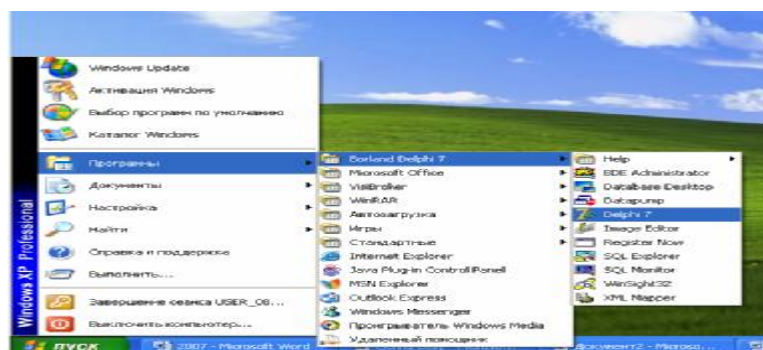
3.1 Test dasturini yaratish algoritmi.

Delphi visual dasturlash tilida obyektga mo'ljallangan dasturlar tuziladi. Bularga misol uchun kalkulyator dasturi, test dasturi xamda mini o'yinlar tuzish mumkin. Test dasturini yaratish uchun biz quyidagi algoritm bo'yicha yaratamiz:

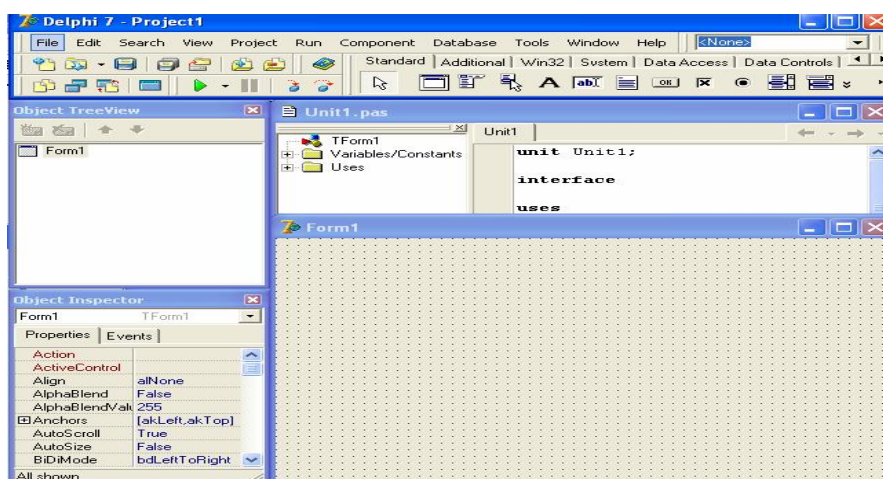
- Delphi dasturi ishga tushiriladi;
- Test dasturining loyihasi tuziladi;
- Loyihaga kerakli komponentalar joylashtiriladi;
- Komponentalarga kerakli kodlar tuziladi;
- Tuzilgan loyiha F9 tugma bilan ishga tushiriladi.

Delphi dasturini ishga tushirish quyidagicha bajariladi:

Пуск => Программы => Borland Delphi => Delphi 7.



Delphi dasturi ishga tushgandan so'ng quyidagi oynaga ega bo'lgan ko'rinish paydo bo'ladi:

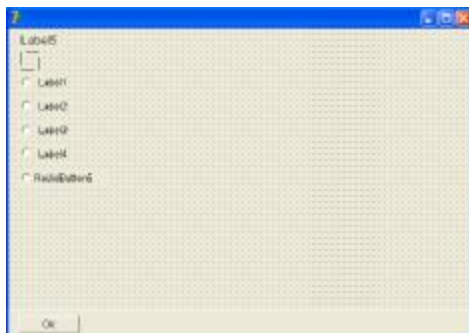


3.2 Dasturda ishlatiladigan komponentalar

Delphi oynasi ishga tushirilgandan so'ng formaga ya'ni Form1 ga test dasturi uchun kerakli komponentalar joylashtiriladi. Komponentalarni Delphining 1 chi oynasi hisoblanmish Delphi Project1 oynasidan olinadi. Bizga quyidagi komponentalar olinadi:

-  5 ta Label komponentasi;
-  1 ta Button komponentasi;
-  5 ta RadioButton komponentasi;
-  1 ta Panel komponentasi loylashtiriladi.

Bu komponentalar Form1 ga quyidagi tartibda joylashtiriladi:



Bu erda bizda Label5 savollar qatori vazifasini bajaradi, Label1, Label2, Label3, Label4 va RadioButton1, RadioButton2, RadioButton3, RadioButton4 xamda RadioButton5 javoblar varianti vazifasini bajaradi. Panel1 esa Button1 komponentasini joylashtirish uchun ishlatiladi. Button1 komponentasi “boshlash” va “keyingi test” ga o’tish buyrug’i vazifasini bajaradi.

3.3 Test dasturining komponentalarida kodlar.

Button tugmasida quyidagi kodlar kiritiladi:

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

```
begin
```

```
  case Button1.tag of
```

```
    0: begin
```

```
      Button1.caption:='Keyingi';
```

```
      Button1.tag:=1;
```

```
      RadioButton5.Checked:=TRUE;
```

```
      Button1.Enabled:=False;
```

```
      ResetForm;
```

```
      VoprosToScr;
```

```
    end;
```

```
    1: begin
```

```
      summa:=summa+score[otv];
```

```
      RadioButton5.Checked:=TRUE;
```

```
      Button1.Enabled:=False;
```

```
      ResetForm;
```

```
      if not eof(f)
```

```
        then VoprosToScr
```

```
        else
```

```
          begin
```

```
            summa:=summa+score[otv];
```

```

        closefile(f);
        Button1.caption:='Ok';
        Form1.caption:='Natija';
        Button1.tag:=2;
        Button1.Enabled:=TRUE;
        Itog;
    end;
end;
2: begin
    Form1.Close;
end;
end;
end;

```

Form1 ning kod ko'rinishi quyidagicha :

```

procedure TForm1.FormCreate(Sender: TObject);
begin
    Image1.AutoSize := False;
    Image1.Proportional := True;
    RadioButton1.Visible := False;
end;
end.

```

RadioButton5 ning kodi quyidagicha:

```

procedure TForm1.RadioButtonClick(Sender: TObject);
begin
    if sender = RadioButton1
    then otv:=1
    else if sender = RadioButton1
    then otv:=2
    else if sender = RadioButton3
    then otv:=3
    else otv:=4;
    Button1.enabled:=TRUE;
end;

```

Formani activate ko'rinishiga keltirib quyidagi kodlarni tuzamiz:

```

procedure TForm1.FormActivate(Sender: TObject);
begin
    ResetForm;
    if ParamCount = 0
    then begin

```

```

        Label5.caption:= 'Test savolining fayli berilmagan.';
        Button1.caption:='Ok';
        Button1.tag:=2;
        Button1.Enabled:=TRUE
    end
else begin
    fn := ParamStr(1);
    assignfile(f,fn);
    try
        reset(f);
    except
        on EFOpenError do
            begin
                ShowMessage('Test fayli '+fn+' topilmadi. ');
                Button1.caption:='Ok';
                Button1.tag:=2;
                Button1.Enabled:=TRUE;
                exit;
            end;
        end;
    Info;
    GetLevel;
end;
end;
end;

```

ITog o'zgaruvchisini xosil qilib quyidagi kodni tuzamiz :

```

procedure TForm1.ITog;
var
    i:integer;
    buf:string;
begin
    buf:="";
    buf:='Test natijasi'+ #13 + #13
        +'Umumiy ball: '+ IntToStr(summa);
    i:=1;
    while (summa < level[i]) and (i<N_LEV) do
        i:=i+1;
    buf:=buf+ #13+mes[i];
    Label5.Top:=20;
    Label5.caption:=buf;
end;

```

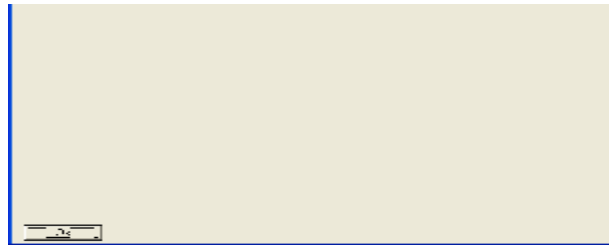
Keyin esa protseduralar kiritiladi:

```

procedure FormActivate(Sender:
TObject);
procedure Button1Click(Sender:
TObject);
procedure RadioButtonClick(Sender: TObject);

procedure Info;
procedure VoprosToScr;
procedure ShowPicture;
procedure ResetForm;
procedure Itog;
procedure FormCreate(Sender: TObject);

```



Quyidagi o'zgarmlar kiritiladi :

```

const
  N_LEV=4;
  N_ANS=4;

```

Quyidagi o'zgaruvchilar kiritiladi :

```

var
  f:TextFile;
  fn:string;

  level:array[1..N_LEV] of integer;
  mes:array[1..N_LEV] of string;

  score:array[1..N_ANS] of integer;
  summa:integer;
  vopros:integer;
  otv:integer;

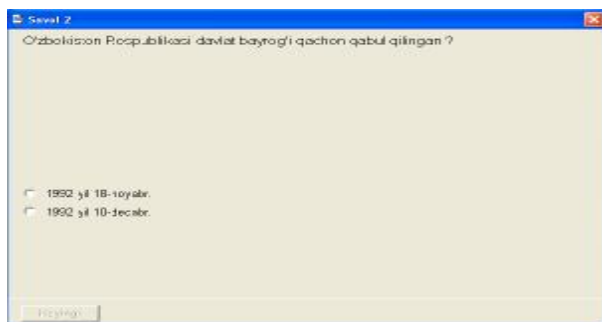
```



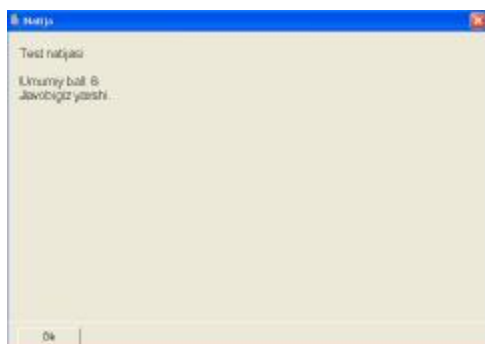
Bu dastur kodlari kiritilgandan so'ng F9 tugmasi bilan ishga tushiriladi, ishga tushirilgan dastur quyidagi ko'rinishga ega bo'ladi:

Bu erda biz Ok tugmasini bosib O'zbekiston Respublikasi xaqidagi savollar testning birinchi savolini ko'rishimiz mumkin, ya'ni quyidagi ko'rinishga ega bo'ladi:

Bu erda biz to'g'ri javobni belgilab "keyingi" tugmasini bosamiz. Tugmani bosganimizdan so'ng ikkinchi savolga o'tiladi, ikkinchi savol quyidagicha bo'ladi;



Xuddi shu tarzda testimizni echishga davom etamiz. Test natijasi esa test tugagandan so'ng ekranga chiqadi, Agar 7 tadan 6 ta to'g'ri ishlansa quyidagi ko'rinishga ega bo'ladi:



Agar 5 ta va undan kam ishlansa quyidagi ko'rinishga ega bo'ladi :



Ok tugmasini bosib testdan chiqish imkonini beradi.

HULOSA

Bu diplom ishda Delphi dasturlash tilida O'zbekiston Respublikasi xaqida "Test" dasturi yaratilgan. Test dasturini yaratishda Delphining 7 chi versiyasidan foydalanilgan. Bu Delphi tilining test dasturi uchun quyidagi komponentalari ishlatilgan:

- Button komponentasi;
- RadioButton komponentasi;
- Panel komponentasi;
- Label komponentasi.

Bu dasturda button komponentasining vazifasi test ni keyingi sahifasiga o'tish imkonini yaratadi. RadioButton komponentasi esa Test variantlarini kiritish uchun ishlatiladi. Label komponentasining vazifasi savollarni kiritish uchun ishlatiladi. Bu dasturda o'quvchi yoki talaba bermalol test dasturini ishga tushirib O'zbekiston Respublikasi xaqida 7 ta test savollaridan iborat testni echish imkonini beradi. Test ishlangandan so'ng test natijasi ishlanganga qarab javobi chiqadi.

Foydalanilgan adabiyotlar.

1. A.Arangel'skiy "Programmirovaniye v delphi 7" Moskva 2004 g.
2. A.Abramov, G.G.Gnezdilova, E.N.Kapustina, M.iSelyun "Zadachi po programmiravaniyu" Moskva 1988 g.
3. Marko Kentu "Delphi4 dlya professionalov" 1999 g.
4. A.Nemnyugin "Turbo Pascal Programmirovaniye na yazike visokogo urovnya" Piter 2003 g.
5. P.Karimov, S.Irisqulov, AIsaboyev "Dasturlash". O`zbekiston 2003 yil.
6. Aripov, A.B.Axmedov, X.Z .Ikramova, R.M.Irmuhamedova, M.V.Sagatov, A.T.Xaydarovm, A.X.Yakubov, M.Z.Yakubova. "Informatika Axborot Texnologiyalari" Toshkent 2003 yil.
7. A.Goncharova "Samouchitel HTML" Piter 2000 g.
8. T.A.Maxarov "Axborot Texnologiyalari" Toshkent 2005 yil.
9. <http://forum.vinograd.ru>