

**O'ZBEKISTON RESPUBLIKASI AXBOROT TEXNOLOGIYALARI VA
KOMMUNIKATSIYALARINI RIVOJLANTIRISH VAZIRLIGI**

**TOSHKENT AXBOROT TEXNOLOGIYALARI UNIVERSITETI
URGANCH FILIALI**

KOMPYUTER INJINIRINGI FAKULTETI

DASTURIY INJINIRINGI KAFEDRASI

C++ DA DASTURLASH FANIDAN

Kurs ishi

MAVZU: KALKULYATOR DASTURINI YARATISH

Bajardi:

911-14 guruh talabasi
Abdullayev Doston

Ilmiy rahbar:

Dasturiy injiniring
kafedraasi
assistenti
Kuryazov D.

MUNDARIJA

KIRISH.....	2
I. NAZARIY QISM.....	4
1.1. Masalaning qoyilishi.....	4
1.2. Kalkulyator dasturi haqida umumiy ma'lumot.....	5
II. ASOSIY QISM.....	7
2.1. Dasturning funksional xarakteristikasi.....	8
2.2. Kalkulyator dasturini yaratishda foydalanilgan Visual Studio komponentalari	9
2.3. Kalkulyator dasturini Visual Studio muhitida C++ tilida yaratish.....	16
III. XULOSA.....	21
IV. FOYDALANILGAN ADABIYOTLAR.....	22
V. ILOVA.....	23

Kirish

Bugungi kunda kompyuter inson faoliyatining turli sohalariga kirib bormoqda. Natijada dasturiy ta'minotga bo'lgan talab tobora kengayib bormoqda. Bu o'z navbatida dasturchilarga bo'lgan talabni oshiradi.

Hozirgi kunda kompyuter va axborot texnologiyalari taraqqiyotida katta yutuqlarga erishilmoqda. Mamlakatimizda xalq xo'jaligi tarmoqlarida: ishlab chiqarish jarayonlari, ofis ishlari, bank operatsiyalari, axborot almashinuv jarayonlari, milliy xizmat ko'rsatish muassasalari, ilmiy tadqiqot ishlari, texnikalarni avtomatik va masofadan boshqarish masalalari, ommaviy axborot vositalari, tibbiyot maskanlaridagi jarayonlarda kompyuterlar va axborot texnologiyalari muvaffaqiyatli qo'llanilmoqda. Hozirgi vaqtda bir qancha dasturlash tillari mavjud bo'lib, ular har xil sohaga doir masalalarni hal qilishda ishlatiladi.

Dasturlash faninining rivojlanishi bevosita kompyuterlarning rivojlanishi bilan bog'liqdir. O'tgan asrning o'rtalarida kompyuterlarning yaratilishi, ular bilan parallel ravishda uning dasturiy ta'minotini ham rivojlanishiga sabab bo'ldi. Kompyuterlarning texnik ta'minotini rivojlanishi bilan, ular uchun turli dasturiy mahsulotlar yaratila boshlangan va yaratilmoqda, hamda kompyuterlarning texnik ta'minoti bilan parallel darajada rivojlanib bormoqda. Dasturiy ta'minotsiz kompyuterlar oddiy qutidan farqi qolmas edi. Hozirgi vaqtda bir qancha dasturlash tillari mavjud bo'lib, ular har xil sohaga doir masalalarni hal qilishda ishlatiladi. C++ dasturlash tili visual dasturlash texnologiyasini qo'llash asosida turli darajadagi qiyinchilikdagi masalalarni tez hal qilish imkonini beradi. Xozirda Axborot texnologiyalari jadal suratlar bilan rivojlanib bormoqda va hayotning barcha sohalariga tadbiiq etilmoqda. Axborot texnologiyalaridan foydalanish foydalanuvchiga ko'pgina qulayliklar yaratib maqsadga tez va sifatli erishishga yordam beradi. Shu jumladan hozirda o'quv jarayoniga Axborot texnologiyalarini keng jalb etish yaxshi natijalarga olib kelishi isbotlangan faktdir.

Kurs ishining birinchi qismida Visual Studio 2012 dasturi imkoniyatlari haqida so'z boradi. Ikkinchi qismida C++ tili operatorlari tavsifi va ularning qo'llanilishi to'g'risida ma'lumot keltirilgan. Ushbu kurs ishida kalkulyator dasturini yaratish uchun kerakli operatorlar va Visual Studioda qo'llanilgan har bir komponentalar xususiyatlari

haqida ma'lumot berilgan.

Visual Studio, Microsoft firmasining mahsuloti hisoblanadi. Bir necha yillardan beri dasturchilarning asosiy ish quroli bo'lib kelmoqda. Bu dastur orqali nafaqat C++ dasturlash tilida, balki boshqa tillar(C#, F#, FoxPro)da ham dastur tuzish mumkin bo'ladi. Shu sababli bu dastur biroz og'irroq hisoblanadi(katta hajmdagi tezkor xotirani talab qiladi). Visual Studio dasturini bir necha hil versiyalari mavjud bo'lib, hozirgi kunda Visual Studioning yangi versiyasi chiqmoqda. Har bir versiyasiga bir necha modullar qo'shilib, dasturni biroz qiyinlashtirmoqda. Shularni inobatga olib, kompyuteringizni tuzilishiga qarab(tezkor xotira hajmi, protsessor,...), kerakli versiyani o'rnatishni maslahat beraman. Zamondan orqada qolmay deb, oxirgi versiyasini qo'yib, kompyuteringizni qiynab qo'ymang.

Visual Studio bu dastur yozish uchun muhit(joy) hisoblanadi. Bu muhitga, siz tuzgan dastur kodlarini kompyuter kodlarga o'tkazib beradigan kompilyator birlashtirilgan bo'ladi. Har bir dasturlash turi uchun alohida kompilyator mavjud bo'lib, ishlatilgan dasturlash tiliga qarab kompilyator ishga tushadi. Visual Studio dasturini o'rnatib, barcha kompilyatorlar ham avtomat o'rnatiladi.

I. NAZARIY QISM

1.1. Masalaning qo'yilishi

Ushbu kurs ishi Kalkulyator dasturini yaratishga qaratilgan bo'lib, bu dasturning vazifasi hisob kitob ishlarini amalga oshirish hisolanadi.

Ushbu dasturiy vositani ishlab chiqishda o'ziga hos talablar mavjud. Bu talablar quyida sanab o'tiladi.

Dasturiy vositani ishlab chiqish uchun masalaning qo'yilishi haqida yoritib beramiz.

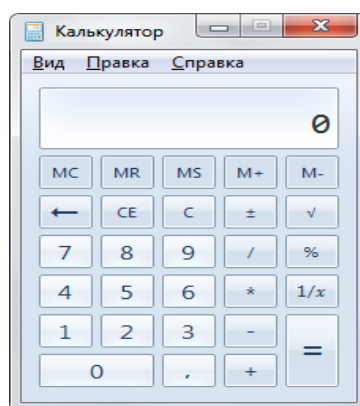
Dastur kompleksi quyidagi vazifalarni bajarishi talab etiladi.

- Arifmetik amallarni bajarilishi;
- Muhandislar uchun amallar ($\sin$, $\cos$, $\tan$);

Ekranda sonlarni kiritish va amallarni tanlash formani oldindan nazarga olish zarur. Menyu qismida muhandislar uchun qo'shimcha amallar kiritilishi zarur.

1.2. Kalkulyator dasturi haqida umumiy malumot

Kalkulyator dasturini ikki xil ko'rinishda foydalanish mumkin: asosiy arifmetik amallarni bajaruvchi standart kalkulyator ko'rinishida va qo'shimcha funktsiyalarni bajaruvchi muxandislik kalkulyatori ko'rinishida. Kalkulyatorni sichqoncha va klaviaturaning yuqori qatoridagi sonlari yoki sonlar blokidagi klaviatura orqali ishlatish mumkin, buning uchun (NumLock tugmasini bosish orqali bajariladi, sonlarni klaviatura orqali kiritish ko'p hisoblash ishlari bajarilganda qulay hisoblanadi).(1.1-rasm)



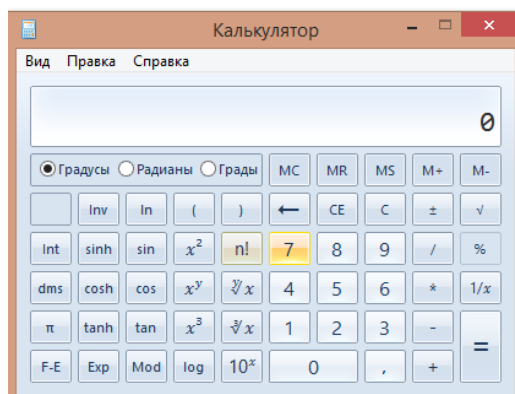
1.1-rasm. Oddiy kalkulyatorni ko'rinishi

Kalkulyator haqida umumiy tushuncha. Bu dastur oddiy kalkulyatorlar kabi amallarni bajaradi. Bu dastur ikki hil rejimda ishlaydi:

- 1) oddiy rejimda;
- 2) Muhandis hisoblashlarni bajarish rejim;

Kalkulyator dasturi quyidagicha ishga tushiriladi. ПУСК + ПРОГРАММЫ + СТАНДАРТНЫЕ + КАЛЬКУЛЯТОР Shundan so'ng Windows ishchi stolida quyidagi oyna hosil bo'ladi: Oddiy hisoblashlar quyidagi ketma-ketlikda bajariladi:

1. Birinchi son kiritiladi.
2. + (qo'shish), - (ayirish), * (ko'paytirish) yoki / (bo'lis) tugmachalari bosiladi..
3. Keyingi son kiritiladi.
4. Qolgan hamma operatorlar va sonlar kiritiladi.
5. = tugmachasi bosiladi Inganer kalkulyator quyidagi ko'rinishga ega:



1.2 –rasm. Muhandislik hisoblashlash kalkulyatori

Muhandislik hisoblashlash quyidagicha amalga oshiriladi; 1. Вид + Инженерный. 2. Sanoq sistemasi tanglanadi. 3. Hisoblashga kirishiladi. Statistik hisoblash quyidagi tartibda bajariladi: 1. Вид + Инженерный. 2. Sonni kiritib Sta tugmacha bosiladi va Статистика oynasi ochiladi. 3. RET tugmachasi bosiladi va kalkulyator oynasiga qaytiladi. Shundan keyin kiritilgan sonni saqlash uchun DAT tugmacha bosiladi. 4. Har bir kiritilgan sondan keyin DAT tugmasi bosiladi. 23 5. Ave, Sum yoki s tugmachalari bosiladi. . Bu yerda : Ave tugmacha yordamida o'rtacha qiymat hisoblanadi; Sum — sonlar yig'indisi hisoblanadi;s — standart siljish hisoblanadi;. Статистика oynasining past qismida hamma saqlangan sonlar ko'rsatilgan. CD tugmachasi yordamida ro'yhatdan ixtiyoriy sonni o'chiriladi, CAD – tugmachasi yordamida hamma sonlar o'chiriladi; LOAD tugmacha yordamida sonni o'zgartirish mumkin.

II. ASOSIY QISM

2.1. Dasturning funktsional xarakteristikasi

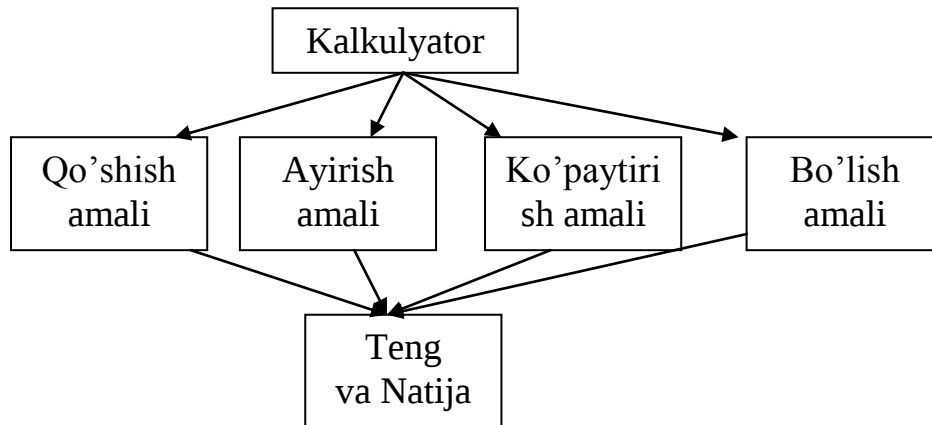
1. Dasturning asosiy vazifalari arifmetik amallarni bajarish.
2. Oldindan ishlab chiqarilgan vaqti noma'lum va hozirda ham keng omma ishlatadi.
3. Shartnoma tuzilmagan.
4. Dasturning zarurligi
Matematik amallarni bajarishda, hisob-kitoblarda ishlatiladi.
5. Dasturning ta'rifi
Qo'shish, ayirish, bo'lish va ko'paytirish amallarini bajaradi.
6. Bu dasturni tuzishdan maqsad va vazifasi hisob kitoblarni tez va xatolarsiz bajarish.
7. Asosiy shart kiritilayotgan sonlar chegaradan oshmasligi va nolga bo'lishni amalga oshirmaslik.
8. Kirish ma'lumotlar sonlar
9. Chiqish ma'lumotlar hisoblangan natija sonlar

Namuna sifatida kalkulyator dasturining berilishi va blok sxemasi.

Dasturni loyihalashda dastur qismlari ishlab chiqiladi. Buni kalkulyator misolida ko'rib chiqamiz.

Eng oddiy dasturlardan biri kalkulyator ishlab chiqish talab qilinsin. Biz osonroq bo'lishi uchun arifmetik amallarni hisoblaydigan kalkulyator interfeysi va uning dasturini ishlab chiqamiz.

Uning funksional sxemasi quyidagicha bo'ladi:



1-sxema. Kalkulyatorning funksional ko'rinishi

Misolni yechishdan avval misolning maqsadini, nima so'ralyaptganini, nimani hisoblash kerakligi, nimalar ma'lum va nimalar noma'lumligini baholaymiz. Baholab bo'lgandan keyin o'z imkoniyatlarimizni baholab qaysi dasturlash tilida, muxitda, proyekt turini va algoritmini aniqlab qaror qabul qilamiz. Ushbu qarorimiz asosida dasturni yozamiz, noma'lum qiymatlarni o'zlashtirish, hisob kitob qiymatlarini ekranda chop ettirish va foydalanuvchi bilan muloqotni o'rnatib, dasturni ishlashini tekshiramiz.

Visual Studio 2012 (VS2012) dasturida dasturlash o'ziga yarasha qulayliklarni yaratadi. Shuning uchun biz shu muhitda dasturlash tilini visual C++ da dasturlashni amalga oshiramiz. Siz esa o'zingizga yoqqan muxitda, dasturlash tilida va albatta o'qituvchining talablarini inobatga olgan holda ishlashingiz mumkin. VS2012 dasturini ishga tushiramiz va VS2012 dasturining asosiy oynasi va bosh sahifa ochiladi (2.1 - rasm). VS2012 dasturida ko'p dasturlash tillari yordamida dasturlash mumkin misol uchun C#, C++, J#, VB va boshqa dasturlash tillari yordamida dasturlash mumkin. Dasturlashni boshlashdan oldin yangi proyekt yoki yaratilgan proyektni ochib dasturlashimiz mumkin. Biz bu misolda yangi proyekt yaratib dasturlashni boshlaymiz.

2.2. Kalkulyator dasturini yaratishda foydalanilgan Visual Studio komponentalari

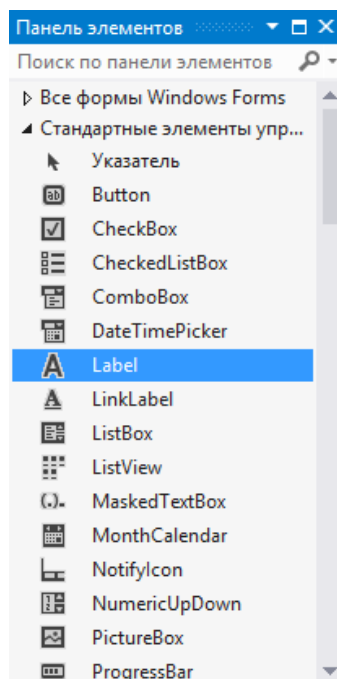
Label komponentasi



Label Yozuv (label komponenti) yorliqni ifodalaydi va ko'pincha o'zining

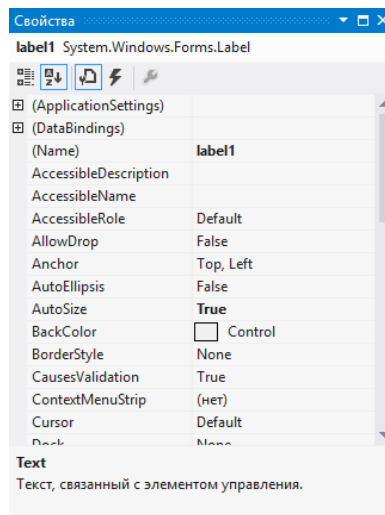
Caption xususiyatiga ega bo`lmagan boshqa boshqaruv elementlarining sarlavasi sifatida ishlatiladi. Ko`pincha yozuvlarni tasvirlash uchun nishon deb nomlanadigan Label komponenti ishlatiladi. U dastur bajarilish vaqti foydalanuvchi tahrirlashi mumkin bo`lgan oddiy matnni tasvirlash uchun ishlatiladi.

Visual Studio dasturida Label komponentasini ishlatish uchun “Панель элементов” oynasida Label nomli komponenta tanlanadi.



2.1-rasm. Label komponentasini ochish

Label komponentining xususiyatlari(Properties) (2.2-rasm)da ko`rsatilgan.



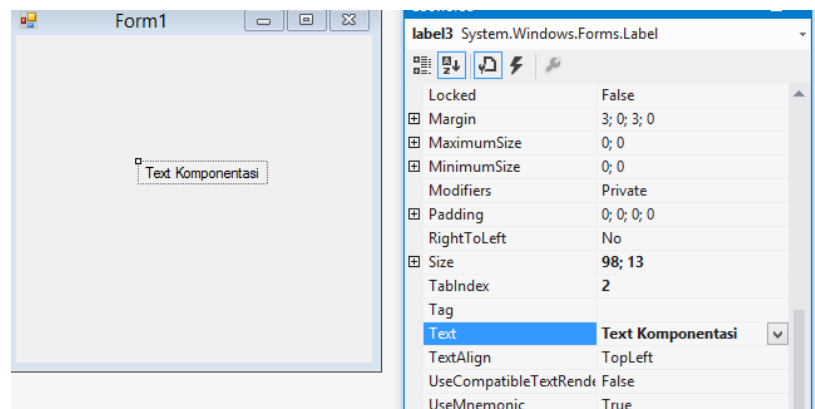
2.2-rasm. Label komponentining xususiyatlari

Bu xususiyatlardan bir nechtasini ko`rib chiqamiz.

BorderStyle — bu xususiyat orqali Label komponentasini chegara sohasini o`zgartirish mumkin.

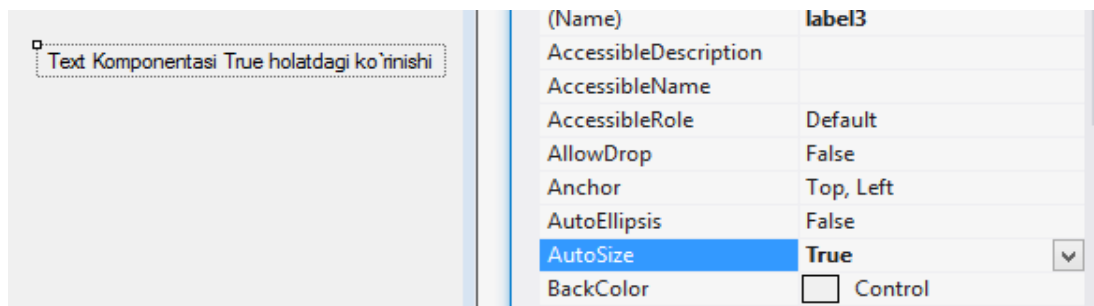
Chegara sohasining uch xil ko`rinishi mavjud: none, (bir chiziqli) окантовка одной линией , окантовка под трехмерное пространство(uch o`lchovli);

Text — xususiyatida Label komponentiga kiritiladigan matn yoziladi.(2.3-rasm)

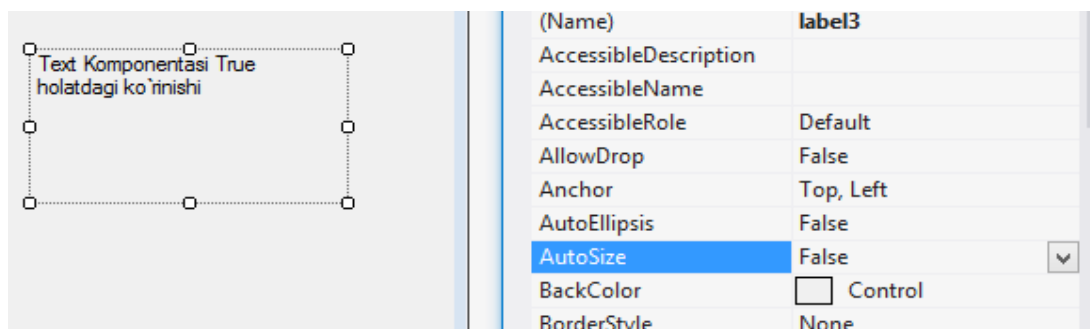


2.3-rasm. Text xususiyati bilan ishlash

AutoSize — Nishon matniga bog`liq ravishda Label komponenti o`lchamlarini avtomatik rostdlashni boshqarish uchun Boolean turidagi AutoSize xususiyati xizmat qiladi. Agar xususiyat (odatdagidek) True qiymatga ega bo`lsa, Label komponenti Caption xususiyatidagi matnga ko`ra o`lchamlarini o`zgartiradi.(2.4,2.5-rasmlar)



2.4-rasm. AutoSize xususiyati bilan ishlash

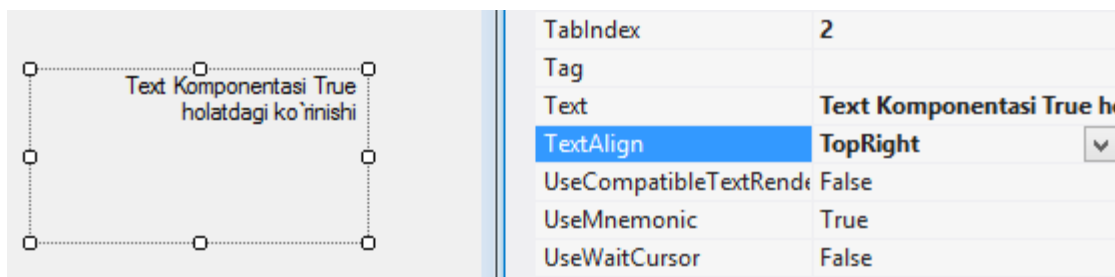


2.5-rasm. AutoSize xususiyati bilan ishlash

TextAlign — yozilgan matnning Label chegarasi bo`ylab qanday joylashishini belgilash. Label komponenti ichida matnni rostdash uchun quyidagi qiymatlarni qabul qiluvchi TAlignment turidagi Alignment xususiyati qo`llaniladi:

- TopLeft– chap yon bo`ylab rostdash;
- TopCenter – matnni markazlashtirish;
- TopRight – o`ng yon bo`ylab rostdash.

Nishonning shoffof yoki bo`yalganligi Boolean turidagi Transporent xususiyati belgilaydi. Bo`yoq rangi Color xususiyati yordamida o`rnatiladi. Odatda, Transporent xususiyati False qiymatiga ega va nishon noshaffof bo`ladi. Shaffof Label komponenti nishon rasm ustida joylashtirilganda va tasvirni yopmasligi zarur bo`lgan hollarda kerak bo`lishi mumkin. Masalan, geografik xaritada. (2.6-rasm)



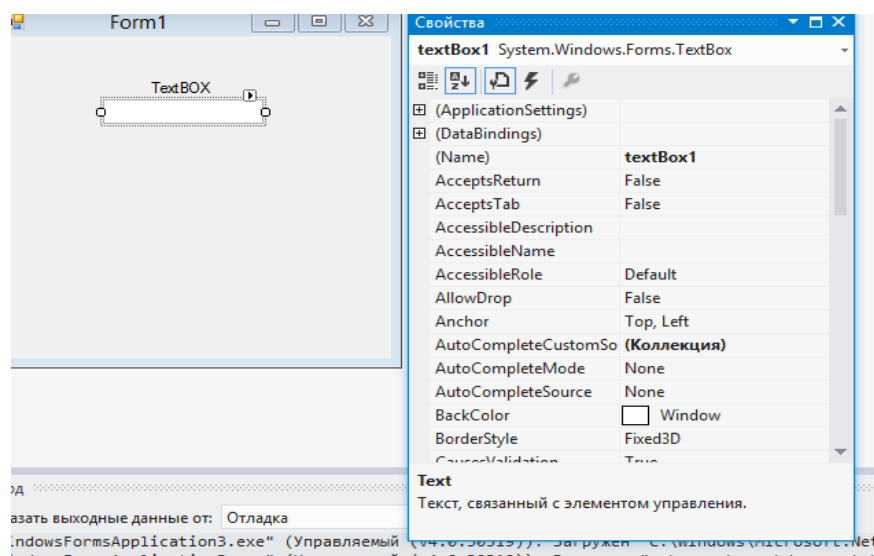
2.6-rasm TextAlign xususyati bilan ishlash

TextBox komponentasi



Bir satrli redaktor matn tasvirlanishi va o'zgartirilishi mumkin bo'lgan ma'lumot kiritish maydonini ifodalaydi. Visual Studioda bir qancha bir satrli redaktorlar mavjud bo'lib, ulardan TextBox komponenti ko'p qo'llaniladi.

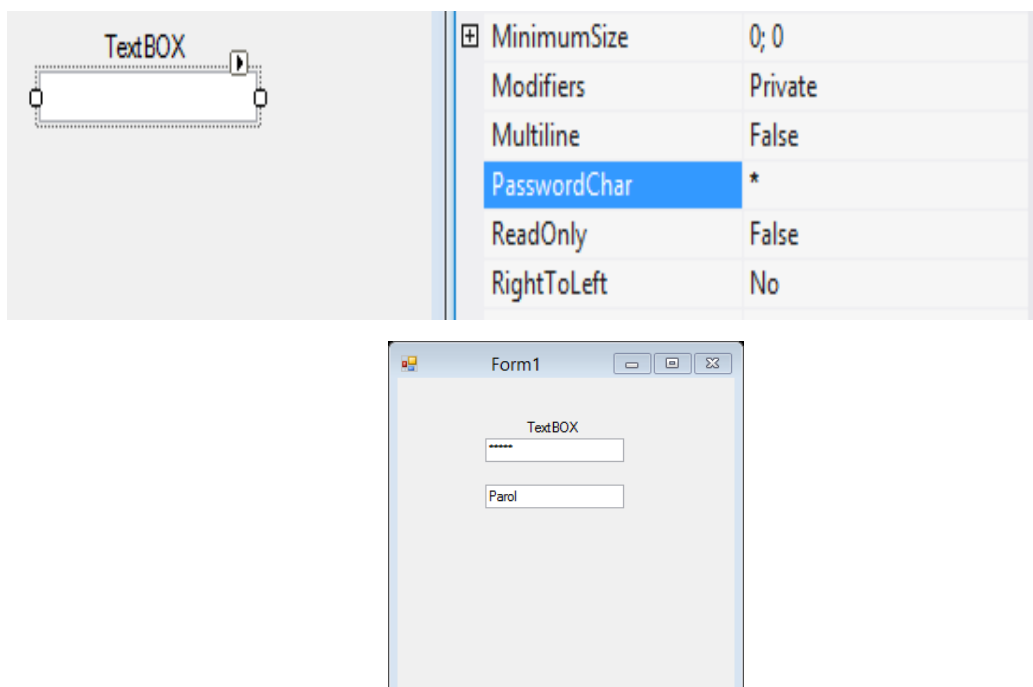
TextBox komponenti klaviaturadan turli simvollarni kiritish va tahrirlashga imkon beradi. Bunda boshqaruv tugmalaridan foydalanib satr bo'ylab siljish, <BackSpace> va <Delete> tugmalari yordamida simvollarni o'cherish, matn (bo'lagini) qismini belgilash va boshqa amallarni bajarish mumkin. Ta'kidlash joizki, bir satrli redaktorlar <Enter> va <Esc> boshqaruv tugmalariga javob bermaydi.(2.7-rasm)



2.7-rasm. TextBox komponenti bilan ishlash

TextBox komponentidan parolni kiritishda foydalanganda Char turidagi

PasswordChar xususiyatidan foydalanish mumkin. U kiritish oynasiga tasvirlanadigan simvolni belgilaydi. Bu simvol matn kiritayotganda haqiqatda kiritilgan ma'lumot o'rnida paydo bo'ladi. Masalan, quyidagi(2.8-rasm)

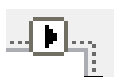


2.8-rasm. PasswordChar xususiyatidan foydalanish

Operatorlari bajarilganda so'ng (tahrirlash) redaktor satrida ***** satri paydo bo'ladi. Aslida esa Text xususiyati Parol qiymatga ega bo'ladi.

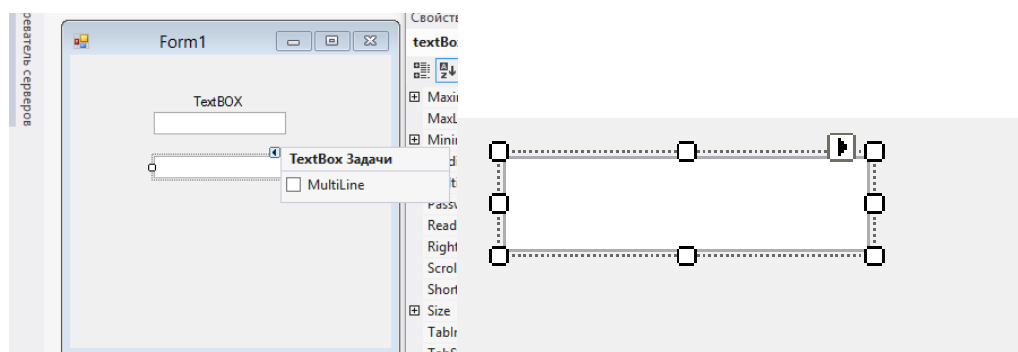
Odatda PasswordChar xususiyati #0 qiymatga ega bo'ladi va redaktor satrida real (haqiqiy) kiritilgan ma'lumot tasvirlanadi.

TextBox komponentasi xususiyatlari



Textbox komponentida o'ng tomonda joylashgan uchurchak belgisiga bosilsa(2.3-rasm)da ko'rsatilgan kichik oyna chiqadi.

MultiLine – bu xususiyat True holatda bo'lsa TextBox komponentini maydoni rostdashni boshqarish mumkin bo'ladi.



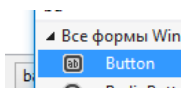
2.9-rasm. MultiLine xususiyati orqali maydoni rostdash

Button komponentasi

Tugmalar boshqaruv elementlari hisoblanadi va maʼlum funksional vazifalarni bajarishga buyruq berish uchun ishlatiladi. Tugma yuzasida matn va yoki rasm tasvirlanishi mumkin.

Visual studio tizimi turli variantdagi tugmalarni tavsiya etuvchi bir qancha komponentalarga ega. Quyidagi turdagi tugmalar mavjud :

- Button standart tugma;
- BitBtn rasmlı tugma;
- SpeedButton tez murojaat tugmasi;



Visual studio standart tugma Button komponenti yordamida berilgan. Tugma yuzasidagi bosilganda bajariladigan ish-harakatni tavsiflovchi yozuvga ega boʻlishi mumkin.

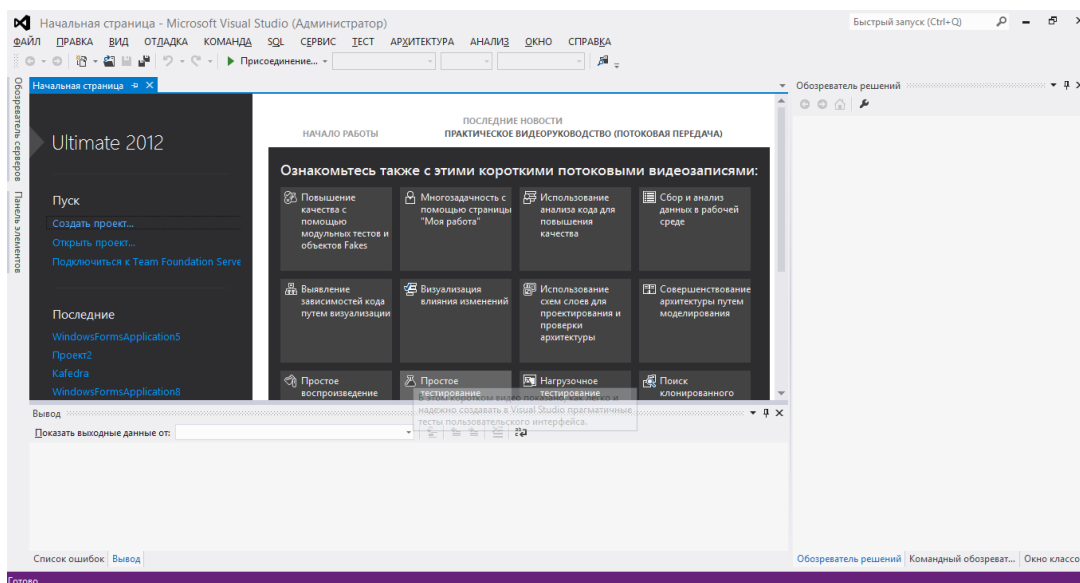
Tugma uchun asosiy hodisa u bosilganda yuzaga keladigan OnClick hodisasi hisoblanadi. Bunda tugma bajarilayotgan harakatni Visual (koʻrinarli) tasvirllovchi mos koʻrinishga ega boʻladi. OnClick hodisasini qayta ishlovchi qism dasturda joylashgan harakat tugma qoʻyib yuborilishini bilanoq bajariladi.

Tugmani quyidagi usullar bilan bosish mumkin:

- sichqonchani qirsillatish bilan ;
- Caption xususiyatida berilgan tugmalar majmuini tanlash bilan;
- <Enter> yoki probel tugmalarini bosish bilan;
- <Esc> tugmasini bosish bilan.

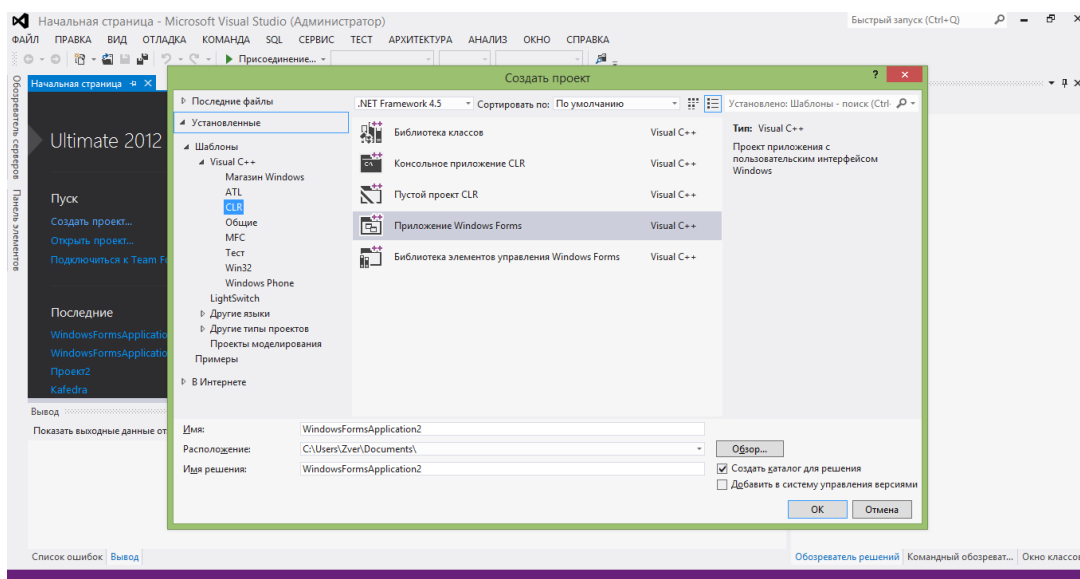
2.3. Kalkulyator dasturini Visual Studio muhitida C++ tilida yaratish

Yangi projektни yaratish fayl menyusи yordamida yoki bosh sahifadan yangi projekt oynasini ochish orqali amalga oshiriladi. Fayl menyusи orqaliochish(2.3.1-rasm)



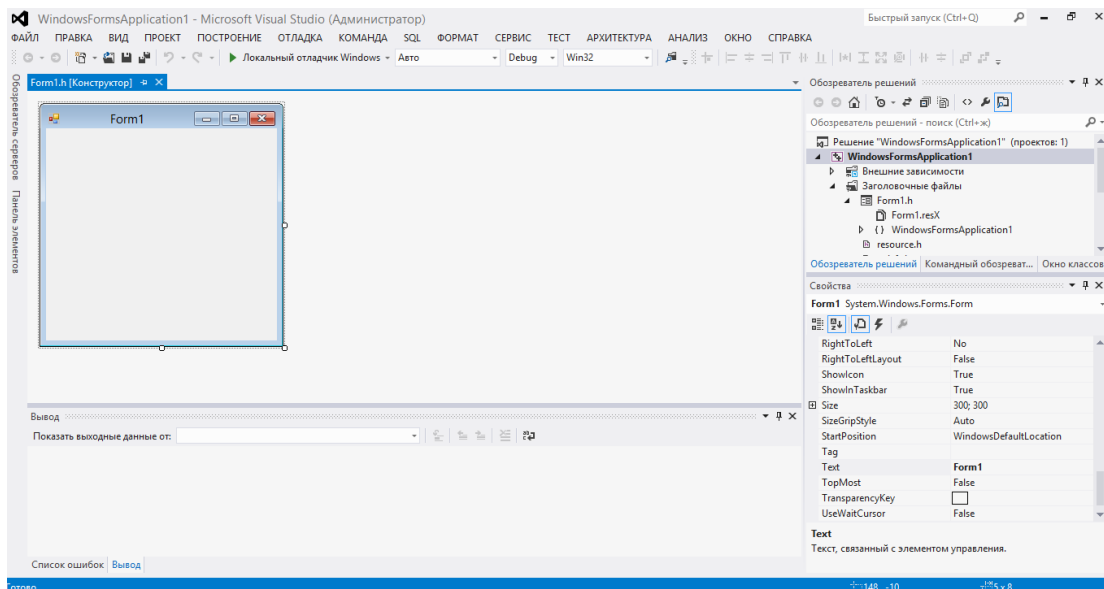
2.3.1 – rasm. Visual Studio bosh saxifasi

File –>New –>Project orqali amalga oshiriladi. Yangi projektни yaratish (2.3.2-rasmda) ko`rsatilgan.



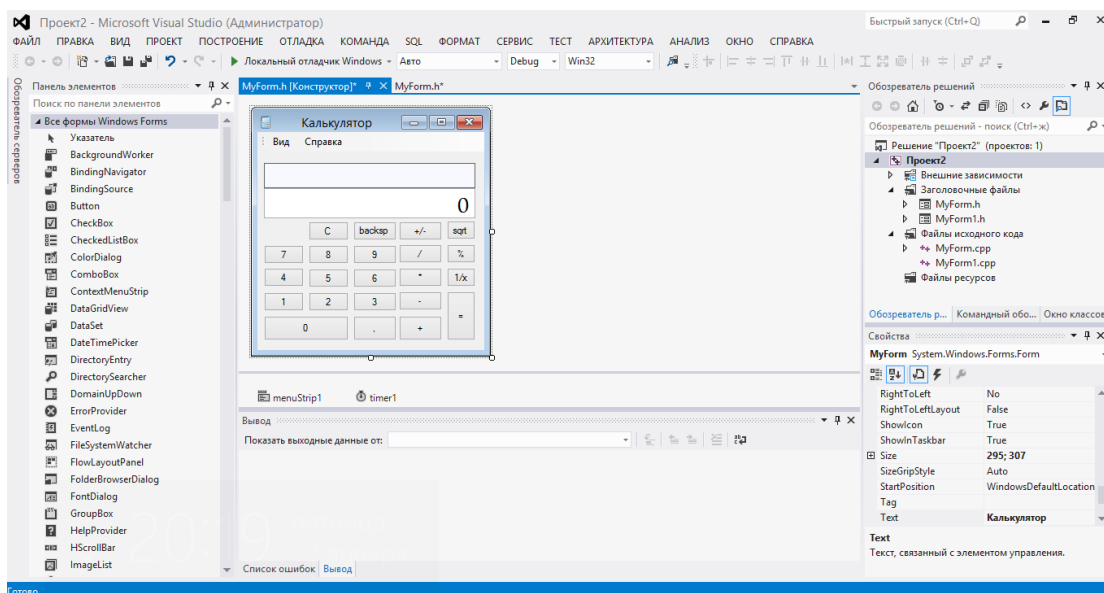
2.3.2 – rasm. Yangi projektни yaratish

Yangi projekt oynasida (2.2 - rasm) dasturlash tilini, projekt turini tanlash, projekt nomini ko`rsatish kerak. Biz dasturlash tilining Visual C++, projektни turi nomini va saqlash joyini ko`rsatamiz, **Ok** tugmasini bosamiz va shunda (2.3.3 – rasmdagi oyna ochiladi).



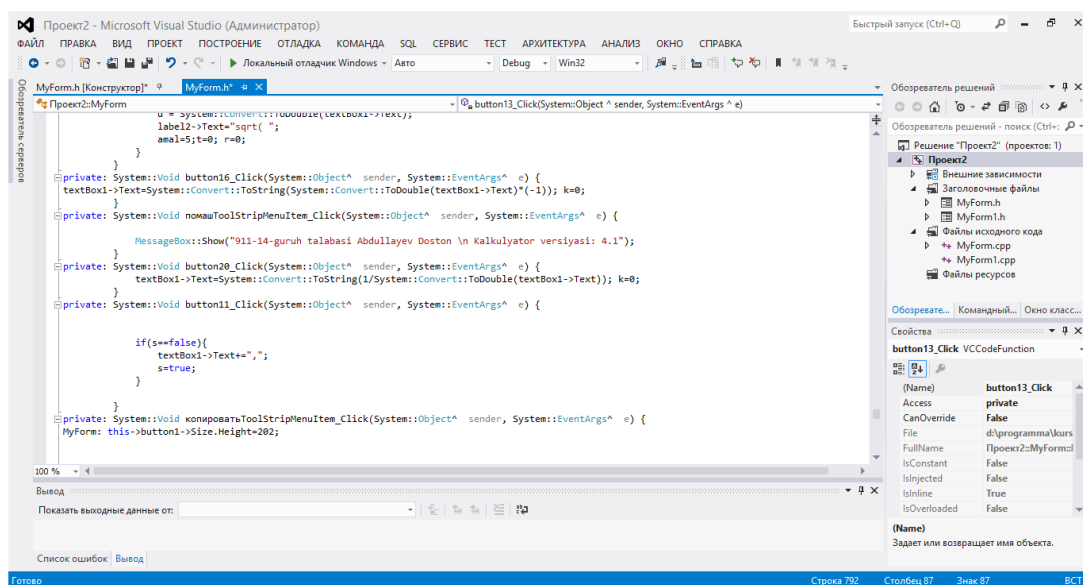
2.3.3 – rasm. Form oynasini ochish

Bu oynada dastur interfeysini yaratamiz va u quyidagicha.



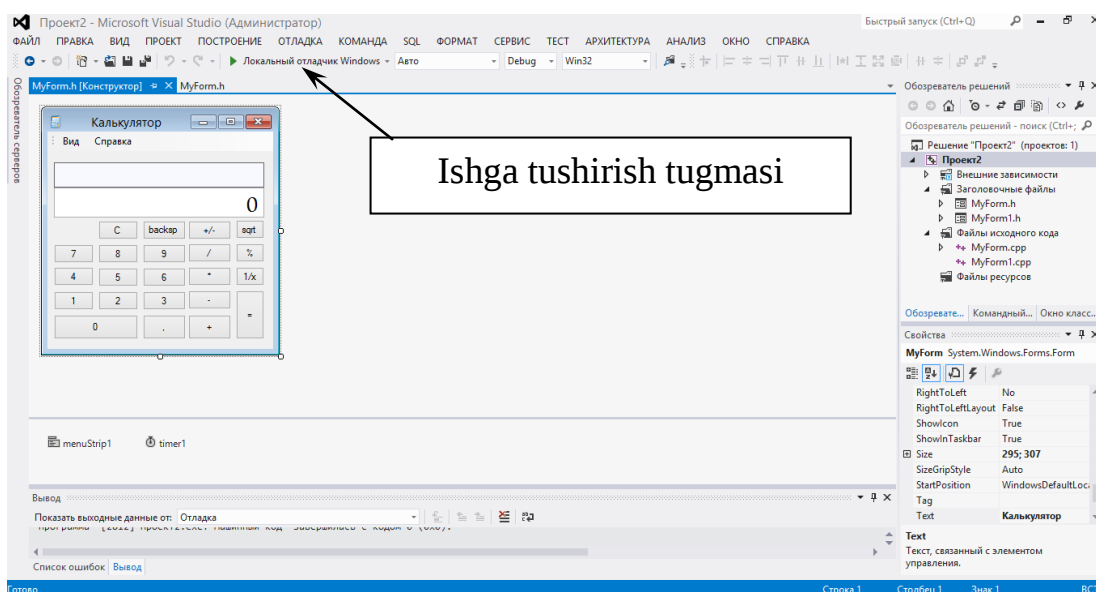
2.3.4– rasm. Kalkulyator interfeysi

Bunda VS2012 ning asosiy oynasi va dasturiy qismi (kodi) saqlanuvchi fayl sahifasi ham ochiladi (2.3.5 – rasm). Bu sahifaga misolning dasturiy qismi yoziladi va shu faylda saqlanadi.



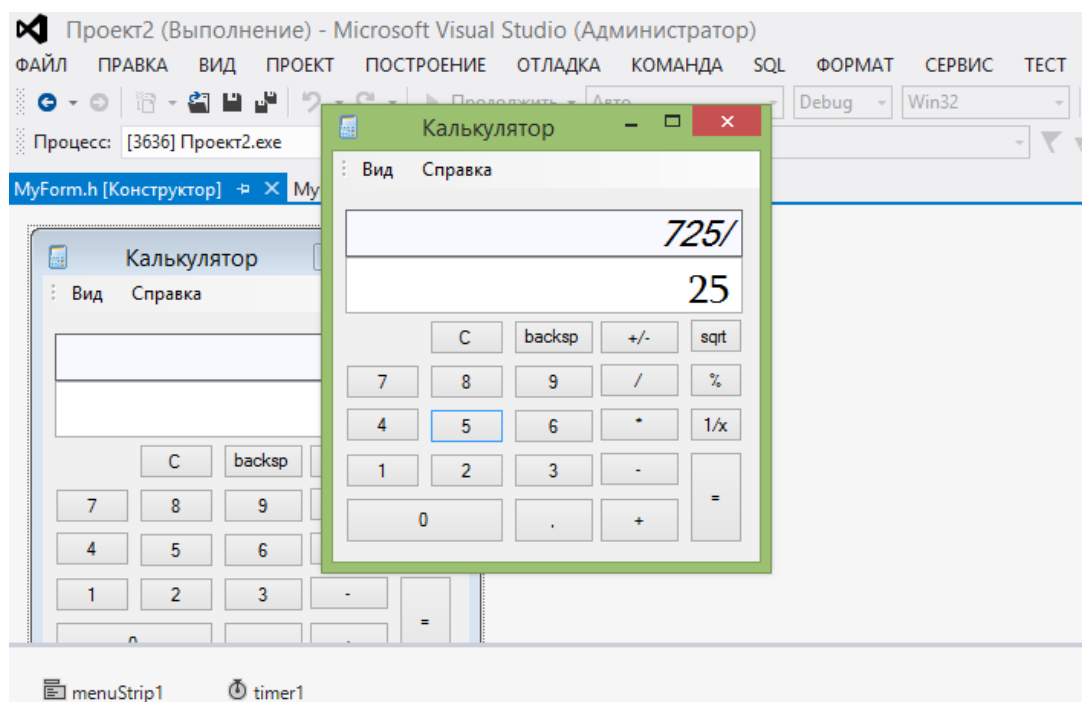
2.3.5 – rasm.Dastur kodi

Dasturlashga o'tishdan oldin uning algoritmini tuzish kerak. Dasturlash algoritmi yuqorida ko'rsatilgandek bo'ladi. Dasturlash algoritmini ko'rib chiqamiz(2.3.6-rasm)



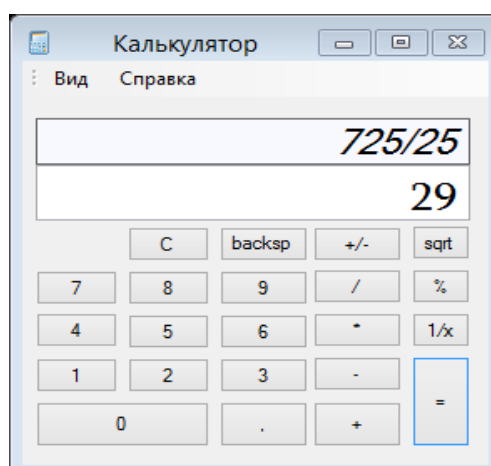
2.3.6– rasm.Dasturni ishga tushirish

Dasturiy qisimni yozib bo'lgandan so'ng dasturni kompilyatsiya qilish va ishga tushirish kerak(2.3.6 – rasm). Kompilyatsiya qilish va ishga tushirish uchun asosiy oynadagi ishga tushirish tugmasi bosiladi va dastur ishga tushadi. Exe fayli ishga tushadi (2.3.7 – rasm) va foydalanuvchi o'rnida o'zim dasturni ishlashini tekshiraman. Ya'ni sonlarni qiymatini klaviatura yordamida kiritaman. Bunda ekranda so'ralgan sonlar kiritilib amal tanlanib teng orqali natija olinadi.



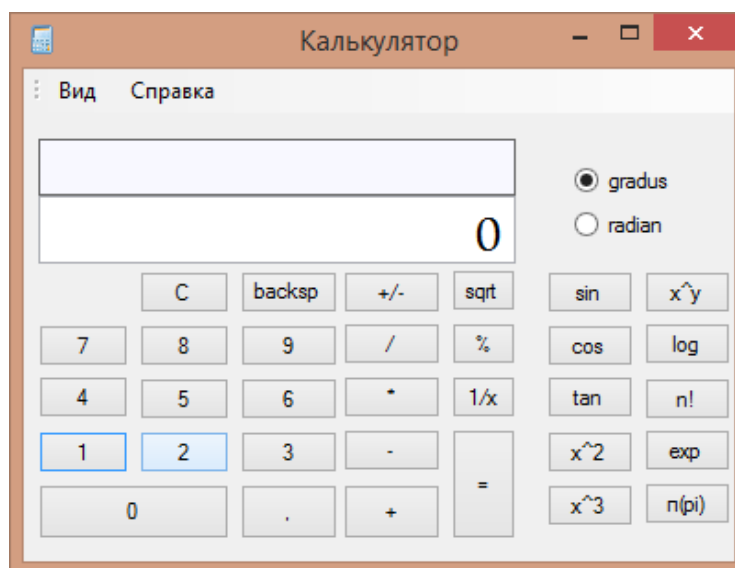
2.3.7 – rasm.Dasturning ishlash jarayoni

Natija bo'linadi va quyidagicha



2.3.8 – rasm.Natija

Muhandislik hisoblashlash quyidagicha amalga oshiriladi:



2.3.8-rasm. Muhandislar uchun qo`shimchalar

1. Вид + Инженерный.
2. Sonni kiritib Sta tugmacha bosiladi va Статистика oynasi ochiladi.
3. RET tugmachasi bosiladi va kalkulyator oynasiga qaytiladi. Shundan keyin kiritilgan sonni saqlash uchun DAT tugmacha bosiladi.
4. Har bir kiritilgan sondan keyin DAT tugmasi bosiladi. 23 5. Ave, Sum yoki s tugmachalari bosiladi. Bu yerda :
Ave tugmacha yordamida o`rtacha qiymat hisoblanadi;

Xulosa

Ushbu dasturiy vositani belgilangan talablar asosida yaratildi. Bu talablar quyida sanab o'tiladi.

Dastur kompleksi quyidagi vazifalarni bajarishi talab etiladi.

- Arifmetik amallarni bajarilishi;
- Muhandislar uchun amallar ($\sin$, $\cos$, $\tan$);

Ekranda sonlarni kiritish va amallarni tanlash formani oldindan nazarga olindi. Menyu qismida muhandislar uchun qo'shimcha amallar kiritildi.

Ushbu yashayotgan zamonamizni kompyuter va texnologik qurilmalarsiz tasavvur qila olmaymiz. Kompyuterlar asosini dasturlar tashkil etadi. Bu kurs ishini ishlash mobaynida biz bunga yana birbora amin bo'ldik. Ushbu kurs ishini tayyorlash mobaynida olgan bilimlarimiz yanada mustahkamlandi. Amaliyotda qo'llab natijalarni ko'rildi. Dasturni tuzishda Visual Studiodan foydalanildi. Bu dasturni Visual Studioning Windows form application kompilyator orqali form yaratib textbox, button va label kompanentalar bilan ishlash o'rganildi.

O'ylaymanki qilgan ishlarimiz kelajakda yurtimizda dasturlash texnologiyasi rivojlanishiga o'z hissasini qo'shadi degan umiddaman.

Foydalanilgan adabiyotlar

1. Борис Пахомов - C,C++ и MS Visual C++ 2012 для начинающих Санкт-Петербург «БХВ-Петербург» 2013
2. Зиборов В. - MS Visual C ++2010 в среде NET Питер, 2012
3. <http://library.ziyonet.uz/ru/book>
4. Khorton_Visual_C__2010_Polny_kurs-2011g Санкт-Петербург 2011
5. Б. Пахомов - C C++ и MS Visual C++ 2008 для начинающих «БХВ-Петербург» 2008
6. Horton I. - Beginning Visual C++ 2012 – 2012
7. Kimmel P., «Borland C++5» . – СПб.: BHV, 1997.
8. Bryan Straustrap. Введение я язык C++ <http://www.infocity.kiev.ua/>
9. Kris Rappas, Uilyam Myurrey. Программирование на C и C++
10. “Ирина”, BHV, Kiev 2000 y.
11. Fridman Aleksandr L’vovich, Язык программирование C++ 2012

V. ILOVA

```
#pragma once
#include<cmath>
namespace Проект2 {

    using namespace System;
    using namespace System::ComponentModel;
    using namespace System::Collections;
    using namespace System::Windows::Forms;
    using namespace System::Data;
    using namespace System::Drawing;

    /// <summary>
    /// Сводка для MyForm
    /// </summary>
    public ref class MyForm : public System::Windows::Forms::Form
    {
    public:
        MyForm(void)
        {
            InitializeComponent();
            //
            //TODO: добавьте код конструктора
            //
        }

    protected:
        /// <summary>
        /// Освободить все используемые ресурсы.
        /// </summary>
        ~MyForm()
        {
            if (components)
            {
                delete components;
            }
        }
    }

#pragma endregion
    int amal;
    Double q,t,a,k,v,r,u;
    bool s;
    double b;
```

```

        void start(){
            if(this->Width<=402)
                this->Width+=4;
        }

        private: System::Void MyForm_Load(System::Object^ sender,
            System::EventArgs^ e) {
            s=false;
        }

        private: System::Void label1_Click(System::Object^ sender,
            System::EventArgs^ e) {
        }

        private: System::Void npbkaToolStripMenuItem_Click(System::Object^ sender,
            System::EventArgs^ e) {
        }

        private: System::Void button1_Click(System::Object^ sender,
            System::EventArgs^ e) {

            if(textBox1->Text=="0")
                textBox1->Text="1";
            else {
                textBox1->Text+="1";
            }
        }

        private: System::Void button2_Click(System::Object^ sender,
            System::EventArgs^ e) {
            if(textBox1->Text=="0")
                textBox1->Text="2";
            else {
                textBox1->Text+="2";
            }
        }

        private: System::Void button3_Click(System::Object^ sender,
            System::EventArgs^ e) {
            if(textBox1->Text=="0")
                textBox1->Text="3";
            else {
                textBox1->Text+="3";
            }
        }

        private: System::Void button6_Click(System::Object^ sender,
            System::EventArgs^ e) {
            if(textBox1->Text=="0")
                textBox1->Text="4";
            else {
                textBox1->Text+="4";
            }
        }

        private: System::Void button5_Click(System::Object^ sender,

```

```

System::EventArgs^ e) {
    if(textBox1->Text=="0")
        textBox1->Text="5";
    else {
        textBox1->Text+="5";
    }
}

private: System::Void button4_Click(System::Object^ sender,
System::EventArgs^ e) {
    if(textBox1->Text=="0")
        textBox1->Text="6";
    else {
        textBox1->Text+="6";
    }
}

private: System::Void button8_Click(System::Object^ sender,
System::EventArgs^ e) {
    if(textBox1->Text=="0")
        textBox1->Text="7";
    else {
        textBox1->Text+="7";
    }
}

private: System::Void button7_Click(System::Object^ sender,
System::EventArgs^ e) {
    if(textBox1->Text=="0")
        textBox1->Text="8";
    else {
        textBox1->Text+="8";
    }
}

private: System::Void button10_Click(System::Object^ sender,
System::EventArgs^ e) {
    if(textBox1->Text=="0")
        textBox1->Text="9";
    else {
        textBox1->Text+="9";
    }
}

private: System::Void button9_Click(System::Object^ sender,
System::EventArgs^ e) {
    if(textBox1->Text=="0")
        textBox1->Text="0";
    else {
        textBox1->Text+="0";
    }
}

private: System::Void button18_Click(System::Object^ sender,
System::EventArgs^ e) {

```

```

        if(textBox1->Text->Length > 0)
            textBox1->Text = textBox1->Text->Substring(0, textBox1->Text->Length-1);
        else
            textBox1->Text = "0";
        if(textBox1->Text->Length == 0) textBox1->Text = "0";
        s=false;
    }
private: System::Void button21_Click(System::Object^ sender,
System::EventArgs^ e) {
    if(r==0){ label2->Text+=textBox1->Text+" ";}
        switch (amal)
        {
            case 1: textBox1->Text=System::Convert::ToString(System::Convert::ToDouble(textBox1->Text)+q); q=0; break;
            case 2: textBox1->Text=System::Convert::ToString(a-System::Convert::ToDouble(textBox1->Text)); a=0; break;
            case 3: textBox1->Text=System::Convert::ToString(System::Convert::ToDouble(textBox1->Text)*k); k=0; break;
            case 4: textBox1->Text=System::Convert::ToString(b/System::Convert::ToDouble(textBox1->Text)); b=0; break;
            case 5: textBox1->Text=System::Convert::ToString(sqrt(u)); u=0; label2->Text+=")"; break;
            case 6: textBox1->Text=System::Convert::ToString(pow(v, System::Convert::ToDouble(textBox1->Text))); v=0;
            default:
                break;
        } r=1;
    }
private: System::Void button17_Click(System::Object^ sender,
System::EventArgs^ e) {
    textBox1->Text="0";
    label2->Text=" ";
    t=0; q=0; a=0; b=0; k=0; u=0;
    s=false;
}
private: System::Void button12_Click(System::Object^ sender,
System::EventArgs^ e) {
    if (q==0) {
        q = System::Convert::ToDouble(textBox1->Text);
        label2->Text=Convert::ToDouble(textBox1->Text)+" ";
        textBox1->Text="0";
        amal=1; t=0;
        r=0;
        s=false;
    }
}

```

```

    }

    }

private: System::Void button15_Click(System::Object^ sender,
System::EventArgs^ e) {
    if (a==0) {
        a = System::Convert::ToDouble(textBox1->Text);
        label2->Text=Convert::ToDouble(textBox1->Text)+"-";
        textBox1->Text="0";
        amal=2; t=0;
        r=0;
        s=false;
    }

    }

private: System::Void button14_Click(System::Object^ sender,
System::EventArgs^ e) {
    if (k==0) {
        k = System::Convert::ToDouble(textBox1->Text);
        label2->Text=Convert::ToDouble(textBox1->Text)+"*";
        textBox1->Text="0";
        amal=3;t=0;
        r=0;
        s=false;
    }

    }

private: System::Void button13_Click(System::Object^ sender,
System::EventArgs^ e) {
    if (b==0) {
        b = System::Convert::ToDouble(textBox1->Text);
        label2->Text=Convert::ToDouble(textBox1->Text)+"/";
        textBox1->Text="0";
        amal=4;t=0; r=0;
        s=false;
    }

    }

private: System::Void exitToolStripMenuItem_Click(System::Object^ sender,
System::EventArgs^ e) {
    timer1->Enabled=false;
    MyForm::Size = System::Drawing::Size(287,296);

    }

private: System::Void button22_Click(System::Object^ sender,
System::EventArgs^ e) {

```

```

        if (u==0) {
            u = System::Convert::ToDouble(textBox1->Text);
            label2->Text="sqrt( ";
            amal=5;t=0; r=0;
        }
    }
private: System::Void button16_Click(System::Object^ sender,
System::EventArgs^ e) {
    textBox1-
>Text=System::Convert::ToString(System::Convert::ToDouble(textBox1-
>Text)*(-1)); k=0;
}
private: System::Void nyomToolStripMenuItem_Click(System::Object^ sender,
System::EventArgs^ e) {

    MessageBox::Show("911-14-guruh talabasi Abdullayev Doston \n
Kalkulyator versiyasi: 4.1");
}
private: System::Void button20_Click(System::Object^ sender,
System::EventArgs^ e) {
    textBox1-
>Text=System::Convert::ToString(1/System::Convert::ToDouble(textBox1-
>Text)); k=0;
}
private: System::Void button11_Click(System::Object^ sender,
System::EventArgs^ e) {

        if(s==false){
            textBox1->Text+=",";
            s=true;
        }

    }
private: System::Void копироватьToolStripMenuItem_Click(System::Object^
sender, System::EventArgs^ e) {
    MyForm: this->button1->Size.Height=202;

}
private: System::Void button23_Click(System::Object^ sender,
System::EventArgs^ e) {
    MessageBox::Show("Salom");
}
private: System::Void иToolStripMenuItem_Click(System::Object^ sender,
System::EventArgs^ e) {
    // MyForm::Size = System::Drawing::Size(402, 296);
    if(this->Width<=402)
        {timer1->Enabled=true;
    }
}

```

```

    }
private: System::Void button26_Click(System::Object^ sender,
System::EventArgs^ e) {
    textBox1-
>Text=System::Convert::ToString(pow(System::Convert::ToDouble(textBox1-
>Text),2));
}
private: System::Void button27_Click(System::Object^ sender,
System::EventArgs^ e) {
    textBox1-
>Text=System::Convert::ToString(pow(System::Convert::ToDouble(textBox1-
>Text),3));
}
private: System::Void button23_Click_1(System::Object^ sender,
System::EventArgs^ e) {
    if(radioButton2->Checked==true){
        textBox1-
>Text=System::Convert::ToString(sin(System::Convert::ToDouble(textBox1-
>Text)));
    }if(radioButton1->Checked==true){
        textBox1-
>Text=System::Convert::ToString(sin(System::Convert::ToDouble(textBox1-
>Text)*(Math::PI/180)));
    }}
private: System::Void button24_Click(System::Object^ sender,
System::EventArgs^ e) {
    if(radioButton2->Checked==true){
        textBox1-
>Text=System::Convert::ToString(cos(System::Convert::ToDouble(textBox1-
>Text)));
    }if(radioButton1->Checked==true){
        textBox1-
>Text=System::Convert::ToString(cos(System::Convert::ToDouble(textBox1-
>Text)*(Math::PI/180)));
    }}
private: System::Void button25_Click(System::Object^ sender,
System::EventArgs^ e) {
    if(radioButton2->Checked==true){
        textBox1-
>Text=System::Convert::ToString(tan(System::Convert::ToDouble(textBox1-
>Text)));
    }if(radioButton1->Checked==true){
        textBox1-
>Text=System::Convert::ToString(tan(System::Convert::ToDouble(textBox1-
>Text)*(Math::PI/180)));
    }
}
private: System::Void button29_Click(System::Object^ sender,
System::EventArgs^ e) {
    if(radioButton2->Checked==true){

```

```

        textBox1->Text=System::Convert::ToString(log(System::Convert::ToDouble(textBox1->Text)));
    }if(radioButton1->Checked==true){
        textBox1->Text=System::Convert::ToString(log(System::Convert::ToDouble(textBox1->Text)*(Math::PI/180)));
    }
}
private: System::Void button28_Click(System::Object^ sender,
System::EventArgs^ e) {
    if (v==0) {
        v = System::Convert::ToDouble(textBox1->Text);
        label2->Text=Convert::ToString(textBox1->Text)+"^";
        amal=6;t=0; r=0; textBox1->Text="0";
    }
}
private: System::Void button30_Click(System::Object^ sender,
System::EventArgs^ e) {
    t=1;
    r=System::Convert::ToInt64(textBox1->Text);
    for(int i=1; i<=r; i++){
        t*=i;
    }
    textBox1->Text=Convert::ToString(t);
}
private: System::Void button31_Click(System::Object^ sender,
System::EventArgs^ e) {
    textBox1->Text=System::Convert::ToString(exp(System::Convert::ToDouble(textBox1->Text)));
}
private: System::Void button32_Click(System::Object^ sender,
System::EventArgs^ e) {
    textBox1->Text="3,14159265358979323846";
}
private: System::Void timer1_Tick(System::Object^ sender,
System::EventArgs^ e) {
    start();
}
};
}

```

