Федеральное государственное бюджетное образовательное учреждение высшего образования

РОССИЙСКАЯ АКАДЕМИЯ

НАРОДНОГО ХОЗЯЙСТВА и ГОСУДАРСТВЕННОЙ СЛУЖБЫ

ПРИ ПРЕЗИДЕНТЕ РОССИЙСКОЙ ФЕДЕРАЦИИ

ЭКОНОМИЧЕСКИЙ ФАКУЛЬТЕТ

Направление «Прикладная информатика»

Кафедра «Системного анализа и информатики»

Лабораторная работа №1

ПО дисциплине \_*Объектно-Ориентированное программирование\_\_\_\_\_\_\_\_*\_\_\_\_\_\_\_\_\_\_\_\_\_

Название \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*Калькулятор систем счисления (Calcutron)\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

Студента *\_\_3*\_\_ курса

Группа № \_\_*31-15*\_\_\_\_\_\_\_\_

Форма обучения \_\_*очная*\_\_

\_\_*Томаев*\_\_\_\_\_\_ Фамилия

\_\_*Карим*\_\_\_\_\_\_\_ Имя

\_\_*Артурович*\_\_\_ Отчество

Проверил \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

МОСКВА

2017

***Содержание:***

1. Введение
2. Скриншоты
3. Код основной части программы

* Классы
* События

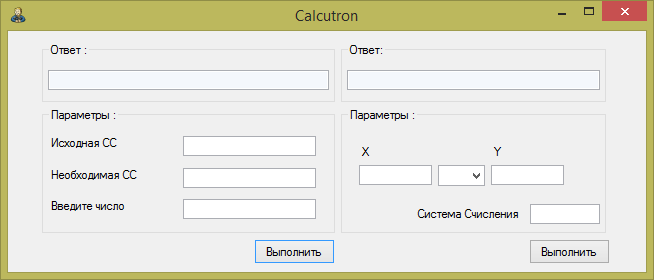
1. Код защитной части программы
2. Дополнение

* Полноценный код для главной формы (За исключением параметров дизайна).

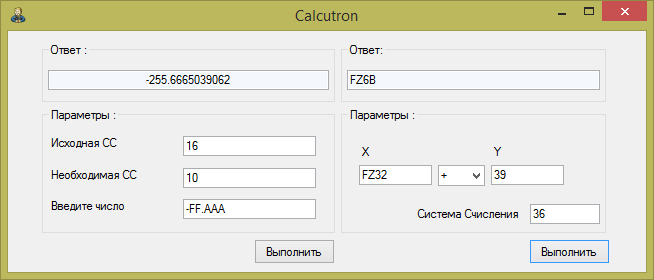
***Введение:***

Цель работы состоит в написании программы, калькулятора, который ,помимо проведения элементарных математических действий, способен переводить числа в различные системы счисления.

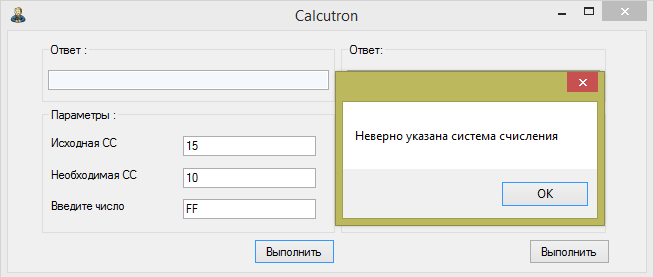
***Скриншоты:***



***Рис. 1***



***Рис. 2***

******

***Рис. 3***

***Код основной части программы:***

***Классы***

**public** static List <**double**> **NumTr**(string ss,string lfa) //Введенное число\строка->List!!!ALAM  
        {  
            var aal =**new** List<**double**>();  
            **for** (**int** i=0;i<ss.Length;i++)  
                aal.**Add**(lfa.**IndexOf**(ss[i]));                  
            return aal;   
        }  
          
        **public** static string **TrNum**(List<**double**>les,string lfa)//List->Num !!!!ALAM превращение результата в вывод  
        {  
            string s1="";  
            **int** Hella=0;  
            **for**(**int** i=0;i<les.Count;i++)  
            {  
                Hella=Convert.**ToInt32**(les[i]);  
                s1=s1+lfa[Hella];  
            }  
            return s1;     
        }  
          
        **public** static **double** **TT**(List<**double**> num,**double** System\_Pas)//To Ten  
        {         
            **double** res = 0 ;  
            **int** gon = 0;  
            **bool** minus=**false**;  
            **if**(num[0]==37)  
            {  
                num.**Remove**(37);  
                minus=**true**;  
            }  
            **int** j = num.Count;  
            **for** (**int** i=0;i<num.Count;i++)  
                **if** (num[i]==36)  
                j=i;  
            **while**(gon!=num.Count)  
            {  
                **if** (num[gon]!=36)//if num[0]=37-bool=true=>0-res  
                {     
                    res=res+num[gon]\*Math.**Pow**(System\_Pas,j-1);        
                    j--;  
                }  
                gon ++;  
            }  
            **if**(minus==**true**)  
                res=0-res;              
            return res;  
        }  
          
        **public** static string **reload**(**double** num,**int** sys,string result,string Alfa)//Tail Trans !!!ALAM result идет снаружи  
        {  
            **double** act;  
            **int** colt;      
            result =".";  
            **for**(**int** i=0;i<10;i++)  
            {  
                act=num\*sys;                  
                colt=Convert.**ToInt32**(act);  
                  
                **if**(colt>act)  
                {  
                    num=act-(colt-1);  
                    result=result+Alfa[colt-1];  
                }  
                **if**(colt<act)  
                {  
                     num=act-colt;  
                     result=result+Alfa[colt];  
                }  
                **if**(colt==act) //если остаток больше числа  
                {  
                    result=result+Alfa[colt];                  
                    break;  
                }  
                  
            }  
            return result;  
              
        }  
          
        **public** static string **FT**(**double** num,**int** System\_Fut,string Lfa)//From Ten no tail  
        {  
            var res =**new** List<**double**>();  
            **int** helpr;  
            string res1="";  
            string res2="";  
            **double** num1=0;  
            **int** colts=Math.**Abs**(Convert.**ToInt32**(num));      
            **double** nummy=0;  
            **if** (num<0)  
            {  
                nummy= num;  
                num=num\*(-1);  
            }  
            **if**(num>0)  
                **if**(colts>num)  
                    colts=colts-1;  
            **if**(num<0)  
                **if**(colts<num)  
                    colts=colts+1;  
            num1=num-colts;  
            num=num-num1;  
            **if**(num==0)  
                res.**Add**(0);  
            **while**(num!=0)  
            {  
                res.**Insert**(0,(num % System\_Fut));  
                helpr=Convert.**ToInt32**(num);  
                num=helpr/System\_Fut;                  
            }  
            **if** (nummy!=0)  
                res.**Insert**(0,37);  
            **for**(**int** i =0;i<res.Count;i++)  
                res1=**TrNum**(res,Lfa);              
            **if**(num1!=0)  
            {  
                res2=**reload**(num1,System\_Fut,res2,Lfa);  
                res1=res1+res2;  
            }  
            return res1;  
        }  
                      
        **public** static string **Act**(string x, string y,string syst,string sign,string Lfa)  
        {  
            var a =**new** List<**double**>();  
            var b =**new** List<**double**>();  
            **double** c;  
            **double** d;  
            **int** System\_Pas=Convert.**ToInt32**(syst);  
            a=**NumTr**(x,Lfa);  
            b=**NumTr**(y,Lfa);  
            c=**TT**(a,System\_Pas);  
            d=**TT**(b,System\_Pas);  
            **switch** (sign)  
            {  
                **case** "+":  
                    c=c+d;  
                    break;  
                **case** "-":  
                    c=c-d;  
                    break;  
                **case** "/":  
                    c=c/d;  
                    break;  
                **case** "\*":  
                    c=c\*d;  
                    break;  
            }  
              
            string result=**FT**(c,System\_Pas,Lfa);  
            return result;  
        }

***События:***

        void **Button1Click**(object sender, EventArgs e)//check  
        {  
            string Numeric="0123456789";  
            string Alfa="0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ.-";  
            **bool** Err = **false**;  
            Err=**Check\_Syst**(*textBox1*.Text,Alfa,Err,Numeric);  
            **if** (Err==**true**)  
            {     
                *textBox1*.**Clear**();  
                *textBox2*.**Clear**();  
                *textBox3*.**Clear**();  
                *textBox4*.Text="Error";  
                return;  
            }  
            Err=**Check\_Syst**(*textBox2*.Text,Alfa,Err,Numeric);  
            **if** (Err==**true**)  
            {     
                *textBox1*.**Clear**();  
                *textBox2*.**Clear**();  
                *textBox3*.**Clear**();  
                *textBox4*.Text="Error";  
                return;  
            }  
            Err=**Check\_Count**(*textBox3*.Text,Alfa,Err,*textBox1*.Text,Numeric,1);  
            **if** (Err==**true**)  
            {     
                *textBox1*.**Clear**();  
                *textBox2*.**Clear**();  
                *textBox3*.**Clear**();  
                *textBox4*.Text="Error";  
                return;  
            }  
            Err=**Check\_Count**(*textBox3*.Text,Alfa,Err,*textBox2*.Text,Numeric,2);  
                  
            **if** (Err==**true**)  
            {     
                *textBox1*.**Clear**();  
                *textBox2*.**Clear**();  
                *textBox3*.**Clear**();  
                *textBox4*.Text="Error";  
                return;  
            }  
            var Num =**new** List<**double**>();  
            **double** D\_Num;      
            **int** Sys\_Pas=Convert.**ToInt32**(*textBox1*.Text);  
            Num=**NumTr**(*textBox3*.Text,Alfa);  
            D\_Num=**TT**(Num,Sys\_Pas);  
            Sys\_Pas=Convert.**ToInt32**(*textBox2*.Text);  
            *textBox4*.Text=**FT**(D\_Num,Sys\_Pas,Alfa);  
              
              
        }  
        void **Button2Click**(object sender, EventArgs e)  
        {  
            **bool** Err = **false**;  
            string Numeric="0123456789";  
            string Alfa="0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ.-";  
            Err=**Check\_Syst**(*textBox8*.Text,Alfa,Err,Numeric);  
            **if** (Err==**true**)  
            {     
                *textBox5*.**Clear**();  
                *textBox6*.**Clear**();  
                *textBox8*.**Clear**();  
                *textBox7*.Text="Error";  
                return;  
            }  
            Err=**Check\_Count**(*textBox5*.Text,Alfa,Err,*textBox8*.Text,Numeric,1);  
            **if** (Err==**true**)  
            {     
                *textBox5*.**Clear**();  
                *textBox6*.**Clear**();  
                *textBox8*.**Clear**();  
                *textBox7*.Text="Error";  
                return;  
            }  
            Err=**Check\_Count**(*textBox6*.Text,Alfa,Err,*textBox8*.Text,Numeric,1);  
                  
            **if** (Err==**true**)  
            {     
                *textBox5*.**Clear**();  
                *textBox6*.**Clear**();  
                *textBox8*.**Clear**();  
                *textBox7*.Text="Error";  
                return;  
            }  
              
            *textBox7*.Text=**Act**(*textBox5*.Text,*textBox6*.Text,*textBox8*.Text,*comboBox1*.Text,Alfa);  
        }  
        void **MainFormShown**(object sender, EventArgs e)  
        {  
              
            *comboBox1*.Items.**Add**("+");  
            *comboBox1*.Items.**Add**("-");  
            *comboBox1*.Items.**Add**("/");  
            *comboBox1*.Items.**Add**("\*");  
        }  
        void **ComboBox1KeyUp**(object sender, KeyEventArgs e)  
        {  
            MessageBox.**Show**("Имеются недопустимые символы");  
            *comboBox1*.Text="";  
      
        }

***Код защитной части программы:***

**bool** **Check\_Syst**(string text,string Lfa,**bool** Er,string num)  
        {   
            **int** b = 0;  
            **bool** n1=**false**;  
            **for**(**int** i=0;i<text.Length;i++)  
              
                **if** (num.**IndexOf**(text[i])==-1)  
                {     
                    MessageBox.**Show**("Имеются недопустимые символы");  
                    Er=**true**;  
                    n1=**true**;  
                    return Er;  
            }  
              
              
            **for**(**int** i=0;i<text.Length;i++)  
            {  
                **if** (Lfa.**IndexOf**(text[i])==-1 || Lfa.**IndexOf**(text[i])>9 )  
                {  
                    MessageBox.**Show**("Имеются недопустимые символы");  
                    Er=**true**;  
                    return Er;  
                }  
            }  
            **for**(**int** i=0;i<text.Length;i++)  
            {  
                **if** (n1==**false**)  
                {  
                b=Convert.**ToInt32**(text);  
                **if** (b>36 || b<2 )  
                {  
                    MessageBox.**Show**("Система счисления выходит за пределы интервалов [2;36]");  
                    Er=**true**;  
                    return Er;  
                }  
                }  
            }  
              
             
             
              
          return Er;  
        }  
          
        **bool** **Check\_Count**(string text,string Lfa,**bool** Er,string Sys,string num,**int** z)  
        {   
            **bool** n1=**false**;  
            **int** f;  
          
            **for**(**int** i=0;i<text.Length;i++)  
            {  
                  
                **if** (Lfa.**IndexOf**(text[i])==-1 )  
                {  
                    MessageBox.**Show**("Имеются недопустимые символы");  
                    Er=**true**;  
                    return Er;  
                      
                }  
                **if** (Lfa.**IndexOf**(text[0])==36 || Lfa.**IndexOf**(text[text.Length-1])==36 || text.**IndexOf**("-") > 0 )  
                {  
                    MessageBox.**Show**("Число записано неверно");  
                    Er=**true**;  
                    return Er;  
  
                }  
                **if** (z==1)  
                {  
                    f=Convert.**ToInt32**(Sys);  
                    **if** (Lfa.**IndexOf**(text[i])>=f && Lfa.**IndexOf**(text[i])!=36 && Lfa.**IndexOf**(text[i])!=37)  
                    {  
                    MessageBox.**Show**("Неверно указана система счисления");  
                    Er=**true**;  
                    return Er;  
                    }  
                }  
                      
            }  
          return Er;  
        }

***Дополнение:***

**using** System;  
**using** System.Collections.Generic;  
**using** System.Drawing;  
**using** System.Windows.Forms;  
  
**namespace** Calcutron  
{  
    /// <**summary**>  
    /// Description of MainForm.  
    /// <**/summary**>  
    **public** partial class MainForm : Form  
    {  
         //Перевод строки в числовой список  
          
        **public** static List <**double**> **NumTr**(string ss,string lfa) //Введенное число\строка->List!!!ALAM  
        {  
            var aal =**new** List<**double**>();  
            **for** (**int** i=0;i<ss.Length;i++)  
                aal.**Add**(lfa.**IndexOf**(ss[i]));                  
            return aal;   
        }  
          
        **public** static string **TrNum**(List<**double**>les,string lfa)//List->Num !!!!ALAM превращение результата в вывод  
        {  
            string s1="";  
            **int** Hella=0;  
            **for**(**int** i=0;i<les.Count;i++)  
            {  
                Hella=Convert.**ToInt32**(les[i]);  
                s1=s1+lfa[Hella];  
            }  
            return s1;     
        }  
          
        **public** static **double** **TT**(List<**double**> num,**double** System\_Pas)//To Ten  
        {         
            **double** res = 0 ;  
            **int** gon = 0;  
            **bool** minus=**false**;  
            **if**(num[0]==37)  
            {  
                num.**Remove**(37);  
                minus=**true**;  
            }  
            **int** j = num.Count;  
            **for** (**int** i=0;i<num.Count;i++)  
                **if** (num[i]==36)  
                j=i;  
            **while**(gon!=num.Count)  
            {  
                **if** (num[gon]!=36)//if num[0]=37-bool=true=>0-res  
                {     
                    res=res+num[gon]\*Math.**Pow**(System\_Pas,j-1);        
                    j--;  
                }  
                gon ++;  
            }  
            **if**(minus==**true**)  
                res=0-res;              
            return res;  
        }  
          
        **public** static string **reload**(**double** num,**int** sys,string result,string Alfa)//Tail Trans !!!ALAM result идет снаружи  
        {  
            **double** act;  
            **int** colt;      
            result =".";  
            **for**(**int** i=0;i<10;i++)  
            {  
                act=num\*sys;                  
                colt=Convert.**ToInt32**(act);  
                  
                **if**(colt>act)  
                {  
                    num=act-(colt-1);  
                    result=result+Alfa[colt-1];  
                }  
                **if**(colt<act)  
                {  
                     num=act-colt;  
                     result=result+Alfa[colt];  
                }  
                **if**(colt==act) //если остаток больше числа  
                {  
                    result=result+Alfa[colt];                  
                    break;  
                }  
                  
            }  
            return result;  
              
        }  
          
        **public** static string **FT**(**double** num,**int** System\_Fut,string Lfa)//From Ten no tail  
        {  
            var res =**new** List<**double**>();  
            **int** helpr;  
            string res1="";  
            string res2="";  
            **double** num1=0;  
            **int** colts=Math.**Abs**(Convert.**ToInt32**(num));      
            **double** nummy=0;  
            **if** (num<0)  
            {  
                nummy= num;  
                num=num\*(-1);  
            }  
            **if**(num>0)  
                **if**(colts>num)  
                    colts=colts-1;  
            **if**(num<0)  
                **if**(colts<num)  
                    colts=colts+1;  
            num1=num-colts;  
            num=num-num1;  
            **if**(num==0)  
                res.**Add**(0);  
            **while**(num!=0)  
            {  
                res.**Insert**(0,(num % System\_Fut));  
                helpr=Convert.**ToInt32**(num);  
                num=helpr/System\_Fut;                  
            }  
            **if** (nummy!=0)  
                res.**Insert**(0,37);  
            **for**(**int** i =0;i<res.Count;i++)  
                res1=**TrNum**(res,Lfa);              
            **if**(num1!=0)  
            {  
                res2=**reload**(num1,System\_Fut,res2,Lfa);  
                res1=res1+res2;  
            }  
            return res1;  
        }  
                      
        **public** static string **Act**(string x, string y,string syst,string sign,string Lfa)  
        {  
            var a =**new** List<**double**>();  
            var b =**new** List<**double**>();  
            **double** c;  
            **double** d;  
            **int** System\_Pas=Convert.**ToInt32**(syst);  
            a=**NumTr**(x,Lfa);  
            b=**NumTr**(y,Lfa);  
            c=**TT**(a,System\_Pas);  
            d=**TT**(b,System\_Pas);  
            **switch** (sign)  
            {  
                **case** "+":  
                    c=c+d;  
                    break;  
                **case** "-":  
                    c=c-d;  
                    break;  
                **case** "/":  
                    c=c/d;  
                    break;  
                **case** "\*":  
                    c=c\*d;  
                    break;  
            }  
              
            string result=**FT**(c,System\_Pas,Lfa);  
            return result;  
        }  
          
        **public** MainForm()  
        {  
            //  
            // The InitializeComponent() call is required for Windows Forms designer support.  
            //  
            **InitializeComponent**();  
              
            //string Alfa="0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ.-";  
            // **TODO**: Add constructor code after the InitializeComponent() call.  
            //  
              
        }  
          
        **bool** **Check\_Syst**(string text,string Lfa,**bool** Er,string num)  
        {   
            **int** b = 0;  
            **bool** n1=**false**;  
            **for**(**int** i=0;i<text.Length;i++)  
              
                **if** (num.**IndexOf**(text[i])==-1)  
                {     
                    MessageBox.**Show**("Имеются недопустимые символы");  
                    Er=**true**;  
                    n1=**true**;  
                    return Er;  
            }  
              
              
            **for**(**int** i=0;i<text.Length;i++)  
            {  
                **if** (Lfa.**IndexOf**(text[i])==-1 || Lfa.**IndexOf**(text[i])>9 )  
                {  
                    MessageBox.**Show**("Имеются недопустимые символы");  
                    Er=**true**;  
                    return Er;  
                }  
            }  
            **for**(**int** i=0;i<text.Length;i++)  
            {  
                **if** (n1==**false**)  
                {  
                b=Convert.**ToInt32**(text);  
                **if** (b>36 || b<2 )  
                {  
                    MessageBox.**Show**("Система счисления выходит за пределы интервалов [2;36]");  
                    Er=**true**;  
                    return Er;  
                }  
                }  
            }  
              
             
             
              
          return Er;  
        }  
          
        **bool** **Check\_Count**(string text,string Lfa,**bool** Er,string Sys,string num,**int** z)  
        {   
            **bool** n1=**false**;  
            **int** f;  
          
            **for**(**int** i=0;i<text.Length;i++)  
            {  
                  
                **if** (Lfa.**IndexOf**(text[i])==-1 )  
                {  
                    MessageBox.**Show**("Имеются недопустимые символы");  
                    Er=**true**;  
                    return Er;  
                      
                }  
                **if** (Lfa.**IndexOf**(text[0])==36 || Lfa.**IndexOf**(text[text.Length-1])==36 || text.**IndexOf**("-") > 0 )  
                {  
                    MessageBox.**Show**("Число записано неверно");  
                    Er=**true**;  
                    return Er;  
  
                }  
                **if** (z==1)  
                {  
                    f=Convert.**ToInt32**(Sys);  
                    **if** (Lfa.**IndexOf**(text[i])>=f && Lfa.**IndexOf**(text[i])!=36 && Lfa.**IndexOf**(text[i])!=37)  
                    {  
                    MessageBox.**Show**("Неверно указана система счисления");  
                    Er=**true**;  
                    return Er;  
                    }  
                }  
                      
            }  
          return Er;  
        }  
          
        void **Button1Click**(object sender, EventArgs e)//check  
        {  
            string Numeric="0123456789";  
            string Alfa="0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ.-";  
            **bool** Err = **false**;  
            Err=**Check\_Syst**(*textBox1*.Text,Alfa,Err,Numeric);  
            **if** (Err==**true**)  
            {     
                *textBox1*.**Clear**();  
                *textBox2*.**Clear**();  
                *textBox3*.**Clear**();  
                *textBox4*.Text="Error";  
                return;  
            }  
            Err=**Check\_Syst**(*textBox2*.Text,Alfa,Err,Numeric);  
            **if** (Err==**true**)  
            {     
                *textBox1*.**Clear**();  
                *textBox2*.**Clear**();  
                *textBox3*.**Clear**();  
                *textBox4*.Text="Error";  
                return;  
            }  
            Err=**Check\_Count**(*textBox3*.Text,Alfa,Err,*textBox1*.Text,Numeric,1);  
            **if** (Err==**true**)  
            {     
                *textBox1*.**Clear**();  
                *textBox2*.**Clear**();  
                *textBox3*.**Clear**();  
                *textBox4*.Text="Error";  
                return;  
            }  
            Err=**Check\_Count**(*textBox3*.Text,Alfa,Err,*textBox2*.Text,Numeric,2);  
                  
            **if** (Err==**true**)  
            {     
                *textBox1*.**Clear**();  
                *textBox2*.**Clear**();  
                *textBox3*.**Clear**();  
                *textBox4*.Text="Error";  
                return;  
            }  
            var Num =**new** List<**double**>();  
            **double** D\_Num;      
            **int** Sys\_Pas=Convert.**ToInt32**(*textBox1*.Text);  
            Num=**NumTr**(*textBox3*.Text,Alfa);  
            D\_Num=**TT**(Num,Sys\_Pas);  
            Sys\_Pas=Convert.**ToInt32**(*textBox2*.Text);  
            *textBox4*.Text=**FT**(D\_Num,Sys\_Pas,Alfa);  
              
              
        }  
        void **Button2Click**(object sender, EventArgs e)  
        {  
            **bool** Err = **false**;  
            string Numeric="0123456789";  
            string Alfa="0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ.-";  
            Err=**Check\_Syst**(*textBox8*.Text,Alfa,Err,Numeric);  
            **if** (Err==**true**)  
            {     
                *textBox5*.**Clear**();  
                *textBox6*.**Clear**();  
                *textBox8*.**Clear**();  
                *textBox7*.Text="Error";  
                return;  
            }  
            Err=**Check\_Count**(*textBox5*.Text,Alfa,Err,*textBox8*.Text,Numeric,1);  
            **if** (Err==**true**)  
            {     
                *textBox5*.**Clear**();  
                *textBox6*.**Clear**();  
                *textBox8*.**Clear**();  
                *textBox7*.Text="Error";  
                return;  
            }  
            Err=**Check\_Count**(*textBox6*.Text,Alfa,Err,*textBox8*.Text,Numeric,1);  
                  
            **if** (Err==**true**)  
            {     
                *textBox5*.**Clear**();  
                *textBox6*.**Clear**();  
                *textBox8*.**Clear**();  
                *textBox7*.Text="Error";  
                return;  
            }  
              
            *textBox7*.Text=**Act**(*textBox5*.Text,*textBox6*.Text,*textBox8*.Text,*comboBox1*.Text,Alfa);  
        }  
        void **MainFormShown**(object sender, EventArgs e)  
        {  
              
            *comboBox1*.Items.**Add**("+");  
            *comboBox1*.Items.**Add**("-");  
            *comboBox1*.Items.**Add**("/");  
            *comboBox1*.Items.**Add**("\*");  
        }  
        void **ComboBox1KeyUp**(object sender, KeyEventArgs e)  
        {  
            MessageBox.**Show**("Имеются недопустимые символы");  
            *comboBox1*.Text="";  
      
        }  
      
          
      
          
          
    }  
}