Державний вищий навчальний заклад

«Прикарпатський національний університет імені Василя Стефаника» Кафедра комп’ютерних наук та інформаційних систем

**ЛАБОРАТОРНА РОБОТА №11**

з предмету «Програмування С#»

# Тема: «Вивчення регулярних виразів»

Виконав:

студент групи КН-2

Гриньків В.І.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

«\_\_\_\_»\_\_\_\_\_\_\_\_\_\_2020р.

к.т.н., доц. Ровінський В.А.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

«\_\_\_\_»\_\_\_\_\_\_\_\_\_\_2020р.

Івано-Франківськ – 2020

Мета: Набути практичного досвіду у складанні регулярних виразів

Код програми:

using System;

using System.Text.RegularExpressions;

using System.Windows.Forms;

namespace Laboratorna\_11

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

static Regex day = new Regex(@"(([1-2][0-9])|([3][0-1])|([1-9]))", RegexOptions.IgnoreCase);

Regex january = new Regex(@"(j(([^a]\*a[^n]\*n)|

([^n]\*n[^u]\*u[^a]\*a[^r]\*r[^y]\*y)|

([^u]\*u[^a]\*a[^y]\*y)|

([^n]\*n[^u]\*u[^r]\*r)|

([^a]\*a[^u]\*u[^r]\*r)|

([^u]\*u[^r]\*r[^y]\*y)|

([^a]\*a[^y]\*y)|

([^a]\*a[^u]\*u[^y]\*y))|

([^j]\*j[^n]\*n[^r]\*r)|

([^a]\*a[^n]\*n[^u]\*u[^r]\*r))", RegexOptions.IgnoreCase);

Regex february = new Regex(@"(f(([^e]\*e[^b]\*b)|

([^a]\*a[^r]\*r[^y]\*y)|

([^b]\*b[^a]\*a[^r]\*r)|

([^e]\*e[^r]\*r[^r]\*r)|

([^e]\*e[^u]\*u[^a]\*a[^y]\*y))|([^f]\*f[^b]\*b[^r]\*r[^r]\*r)|([^e]\*e[^b]\*b[^r]\*r[^u]\*u[^a]\*a[^r]\*r))", RegexOptions.IgnoreCase);

Regex march = new Regex(@"((m(([^a]\*a[^r]\*r[^c]\*c[^h]\*h)|([^a]\*a[^r]\*r[^c]\*c)|([^r]\*r[^c]\*c)|([^a]\*a[^r]\*r)|

([^a]\*a[^r]\*r)|([^a]\*a[^c]\*c[^h]\*h)))|([^a]\*a[^r]\*r[^c]\*c[^h]\*h))", RegexOptions.IgnoreCase);

Regex april = new Regex(@"(a(([^p]\*p[^r]\*r[^i]\*i[^l]\*l)|

([^r]\*r[^i]\*i[^l]\*l)|([^a]\*a[^i]\*i[^l]\*l))|([^p]\*p[^r]\*r[^i]\*i[^l]\*l)|([^r]\*r[^i]\*i[^l]\*l))", RegexOptions.IgnoreCase);

//Regex may = new Regex(@"(m([^a]\*a[^y]\*y)|([^a]\*a[^y]\*y[^y]\*y])|([^a]\*a[^a]\*a[^y]\*y))", RegexOptions.IgnoreCase);

Regex may = new Regex(@"(m([^a]\*a[^y]\*y?)|([^a]\*a?[^y]\*y]))", RegexOptions.IgnoreCase);

Regex june = new Regex(@"(j(([^u]\*u[^n]\*n[^e]\*e)|([^n]\*n[^e]\*e)|([^u]\*u[^n]\*n))|([^u]\*u[^n]\*n[^e]\*e))", RegexOptions.IgnoreCase);

Regex july = new Regex(@"(j(([^u]\*u[^l]\*l[^y]\*y)|([^l]\*l[^y]\*y))|([^u]\*u[^l]\*l[^y]\*y))", RegexOptions.IgnoreCase);

Regex august = new Regex(@"(a(([^u]\*u[^g]\*g[^s]\*s[^t]\*t)|

([^g]\*g[^s]\*s[^t]\*t)|

([^g]\*g[^u]^u[^s]\*s[^t]\*t)|

([^u]\*u[^s]\*s[^t]\*t)|([^u]\*u[^t]\*t))|

([^u]\*u[^g]\*g[^s]\*s[^t]\*t)|([^g]\*g[^u]\*u[^s]\*s[^t]\*t))", RegexOptions.IgnoreCase);

Regex september = new Regex(@"s(([^e]\*e[^p]\*p[^t]^t[^m]\*m[^b]\*b[^r]\*r)|

([^p]\*p[^t]\*t[^m]\*m[^b]\*b[^r]\*r)|

([^t]\*t[^e]\*e[^m]\*m[^b]\*b[^r])|

([^t]\*t[^m]\*m[^b]\*b[^r]r))|

([^t]\*t[^e]\*e[^m]\*m[^b]\*b[^e]\*e[^r]\*r)|([^p]\*p[^e]\*e[^m]\*m[^b]\*b)", RegexOptions.IgnoreCase);

Regex october = new Regex(@"(o(([^c] \* c[^t] \* t) |

([^t] \* t[^e] \* e[^r] \* r) |

([^o] \* o[^e] \* e[^r] \* r) |

([^c] \* c[^b] \* b[^e] \* e) | ([^o] \* o[^e] \* e))) |

([^c] \* c[^t] \* t[^b] \* b[^r] \* r) |

([^c] \* c[^t] \* t[^o] \* o[^b] \* b[^e] \* e)", RegexOptions.IgnoreCase);

Regex november = new Regex(@"(n(([^o]\*o[^v]\*v)|

([^v]\*v[^m]\*m[^r]\*r)|

([^b]\*b[^e]\*e[^r]\*r)|

([^o]\*o[^e]\*e[^m]\*m)|

([^o]\*o[^e]\*e[^e]\*e)))|

([^n]\*n[^v]\*v[^m]\*m[^b]\*b[^r]\*r)|

([^o]\*o[^v]\*v[^e]\*e[^m]\*m[^b]\*b[^e]\*e)", RegexOptions.IgnoreCase);

Regex december = new Regex(@"(d(([^e]\*e[^c]\*c)|

([^b]\*b[^e]\*e[^r]\*r)|

([^c]\*c[^e]\*e[^m]\*m)|

([^e]\*e[^m]\*m[^r]\*r)|

([^e]\*e[^e]\*e[^e]\*e)))|

([^d]\*d[^c]\*c[^m]\*m[^b]\*b[^r]\*r)|

([^e]\*e[^m]\*m[^b]\*b[^e]\*e[^r]\*r)", RegexOptions.IgnoreCase);

public void conv()

{

CorrectTextBox.Clear();

string daytxt = SayTextBox.Text;

Match match = Form1.day.Match(daytxt);

string day;

if (Form1.day.IsMatch(daytxt) == true)

{

day = match.Groups[0].Value;

if (january.IsMatch(daytxt) == true)

{

CorrectTextBox.Text = day + " January";

}

else if(february.IsMatch(daytxt) == true && Convert.ToInt32(day)<29)

{

CorrectTextBox.Text = day + " February";

}

else if(march.IsMatch(daytxt) == true)

{

CorrectTextBox.Text = day + " March";

}

else if(april.IsMatch(daytxt)==true)

{

CorrectTextBox.Text = day + " April";

}

else if(may.IsMatch(daytxt) == true)

{

CorrectTextBox.Text = day + " May";

}

else if(june.IsMatch(daytxt)== true)

{

CorrectTextBox.Text = day + " June";

}

else if(july.IsMatch(daytxt) == true)

{

CorrectTextBox.Text = day + " July";

}

else if (august.IsMatch(daytxt) == true)

{

CorrectTextBox.Text = day + " August";

}

else if(september.IsMatch(daytxt) == true)

{

CorrectTextBox.Text = day + " September";

}

else if(october.IsMatch(daytxt) == true)

{

CorrectTextBox.Text = day + " October";

}

else if(november.IsMatch(daytxt) == true)

{

CorrectTextBox.Text = day + " November";

}

else if (december.IsMatch(daytxt) == true)

{

CorrectTextBox.Text = day + " December";

}

else

{

CorrectTextBox.Clear();

}

}

else

{

CorrectTextBox.Clear();

}

}

private void SayTextBox\_TextChanged(object sender, EventArgs e)

{

conv();

}

}

}

Результат виконання:







