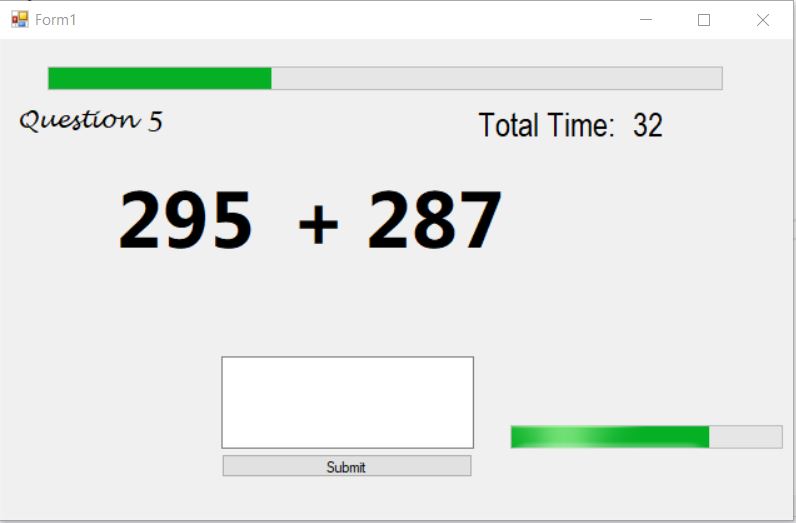
Документација за проектната задача по визуелно програмирање – “Брза математика”

Целта на оваа проектна задача е да се направи забавен квиз кој ќе покаже колку брзо може да пресметувате основни математички изрази. Еве како изгледа играта:

Еве неколку од фукнционалностите на играта што ја прават интересна за играње:

1. Тајмерот за секое наредно прашање станува се пократок. Тоа се прави на тој начин што од прогрес барот во долниот десен агол се одзема различна вредност во зависност од тоа на кое прашање се наоѓате. Ова се имплементира преку овој код:

int a = int.Parse(label5.Text)+1;

int c= progressBar2.Value - a;

if(c<=0)

{

GameOver(getAnswer(), 2);

return;

}

else progressBar2.Value = c;

2. Oперацијата собирање може да се врши на броеви до најмногу 300, одземање до 150 и множење до 15. Ова е за да ја направи играта интересна и да има соодветна тежина.

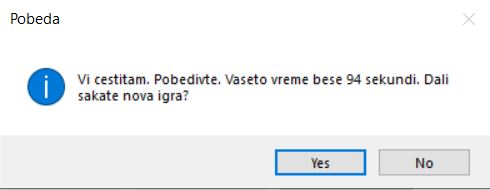
3. Играта завршува во 3 случаи

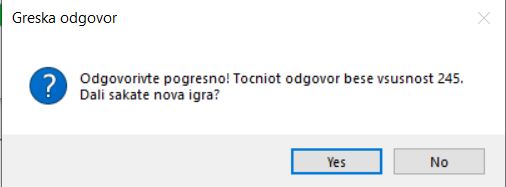
- истечено време

- грешка одговор

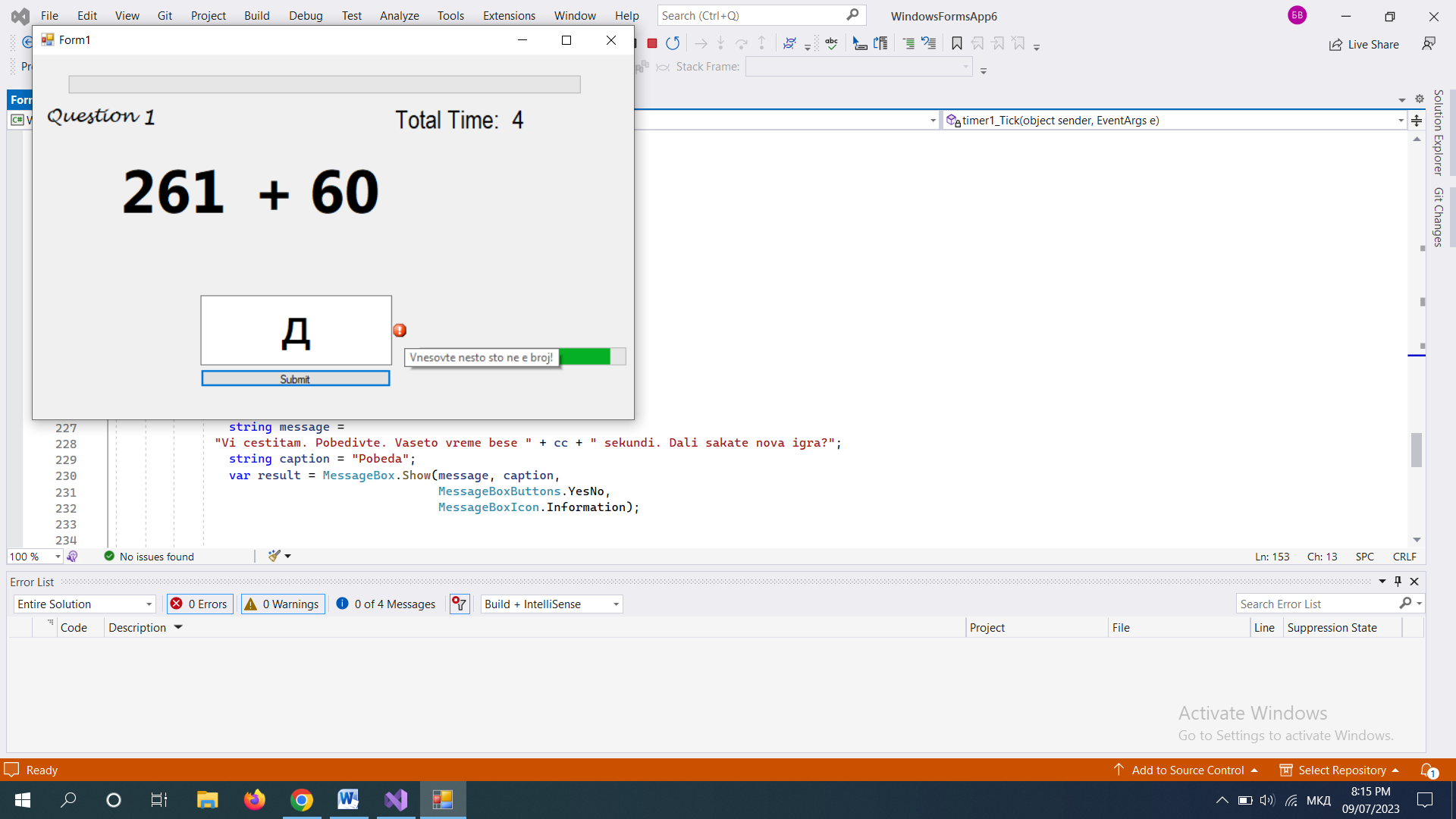
- точно одговорени 15 прашања.

За секој од тие случаи играта испраќа соодветна порака.





Доколку истече времето или е одговорено грешно, се прикажува порака која покажува кој бил точниот одговор, инаку во случај на победа, се прикажува вкупното време во кое сте ја победиле играта.

4. Доколку се внесе нешто што не е број, се јавува Error Provider кој ви кажува дека мора да внесете исклучиво број. 

Опис на две функции:

private void ChangeNumbers()

{

Random r = new Random();

int a, b;

char d;

int c = r.Next(0,3);

if(c==0 || c==1)

{

if (c == 0)

{

a = r.Next(0, 300);

b = r.Next(0, 300);

}

else

{

a = r.Next(0, 100);

b = r.Next(0, 100);

}

if (c==0)

{

d = '+';

}

else

{

d = '-';

}

}

else

{

a = r.Next(0, 20);

b = r.Next(0, 20);

d = '\*';

}

label1.Text = a.ToString();

label3.Text = b.ToString();

label2.Text = d.ToString();

}

Прво се генерира рандом објект. Потоа со помош на тој објект се генерира рандом операција што може да биде собирање, множење или одземање. Тоа се прави така што се генерира случаен број од 0 до 2. 0 означува собирање, 1 одземање, а 2 множење. Со цел играта да има соодветна тежина , доколку е избрано собирање се генерираат броеви меѓу 0 и 300, за одземање меѓу 0 и 100, а за множење меѓу 0 и 20. Потоа соодветните броеви се заменуваат во лабелите и со друга функција се пресметува резултатот.

private int getAnswer()

{

int a = int.Parse(label1.Text);

int b = int.Parse(label3.Text);

int answer;

if (label2.Text == "+")

{

answer = a + b;

}

else if (label2.Text == "-")

{

answer = a - b;

}

else { answer = a \* b; }

return answer;

}

Текстот се претвара во цел број со помош на Parse методот од Integer класата. Потоа во зависност од содржината на средишната лабела label2, се одлучува кој ќе биде одговорот на математичкото прашање.

Изработил Васил Бунгурџиев 211005