

Doc. dr Miroljub Zahorjanski

ZBIRKA ZADATAKA iz programskog jezika C#



Računarski fakultet



CET

Sadržaj:

Uputstvo	11
Uvodni zadaci	13
Zadaci za vežbu	23
Jednostavne forme (1- 12)	
1. Sabiranje dve celobrojne vrednosti iz programskog kôda	23
2. Sabiranje dve celobrojne vrednosti iz TextBox-a	23
3. Kalkulator sa četiri osnovne radnje	23
4. Računanje površine kruga	24
5. Računanje obima kruga	24
6. Računanje kvadratnog korena	24
7. Spajanje reči iz više TextBox-ova	24
8. Izračunavanje broja karaktera u nizu više reči	25
9. Konverzija slova	25
10. Konkatenacija	25
11. Konverzija broja i tekst	25
12. Konverzija teksat u broj	25
Konzolne aplikacije (13 – 21)	
13. Aritmetička sredina	26
14. Lični podaci osobe	26
15. While petlja	26
16. While petlja i zbrajanje	26
17. For petlja i zbrajanje	26
18. For petlja i proizvod	27
19. Testiranje konzolnog prozora	27
20. Prosek unetih brojeva	27
21. Testiranje unetog broja	27
22. Izdvajanje najvećeg unetog broja	28
23. Upis ispitne ocene tekstem na osnovu broja	28
24. Switch petlja i obračun putarine na autoputu	28
25. Računanje prosečne brzine na autoputu	28

Složene forme (22 - 62)

26. Ispis poruke na ekranu	29
27. Otvaranje druge forme iz aktivne forme	29
28. Primena radio dugmeta i prijava ispita	29
29. Primena CheckBox i ankete	29
30. Dijalog poruke i odlučivanje	30
31. PictureBox i učitavanje slike	30
32. Prikazivanje sistemskog datuma i vremena	30
33. Pokretanje eksternih aplikacija	30
34. Kontrola Timer i štoperica	30
35. Validacija unosa u polje na formi	30
36. Konverzija dekadnog u binarni zapis	31
37. Konverzija dekadnog u oktalni zapis	31
38. Konverzija dekadnog u heksadekadni zapis	31
39. Menjanje slike u PictureBox-u iz liste	31
40. Kontrola Timer i odbrojanje	32
41. Validacija unosa u polje na događaj KeyPress	32
42. Ograničenje unosa povodom broja zankova	32
43. Pravljenje tekst editora	32
44. Provera regularnosti unosa u predefinisana polja	33
45. Logovanje iz programskog kôda	33
46. ProgresBar – napredovanje	33
47. Šifriranje i dešifriranje unesenog teksta	34
48. Kontrola TreeView iz programskog kôda	34
49. Štampanje teksta iz polja	34
50. Kontrola dateTimePicker i razlika dva datuma	34
51. Držač mesta TextBox-a	34
52. Upotreba kontrole trackBar (1)	34
53. Upotreba kontrole trackBar (2)	35
54. Kontrola PictureBox i rotiranje slike	35
55. Klizni panel – otkrivanje i sakrivanje kontrole panel	35
56. Datum isteka aplikacije	35
57. Padajući meni	35
58. Traka menija	36
59. Kontekstni meni	36
60. TabControl-e	36
61. Open-Save file dialog	36
62. Višenamenski interfejs	37

Rad sa bazama podataka (63 - 84)

(Projekat „Student”)

63. Provera konekcije	38
64. Punjenje dataGridView podacima iz tabele	38
65. Snimanje novog zapisa	39
66. Prenos fokusa iz kontrole u kontrolu tipkom Enter	39
67. Brisanje zapisa i pražnjenje kontrola za unos	40
68. Sinhronizacija kontrola za unos sa dataGridView kontrolom	40
69. Ažuriranje zapisa	40
70. Osvežavanje prikaza	40

(Projekat „Zaposleni”)

71. Provera konekcije i obrada grške	40
72. Pravljenje forme za logovanje	41
73. Dodavanje slika i ikona na formu i otvaranje druge forme.....	41
74. Inertovanje i snimanje podataka u bazu	41
75. Ažuriranje podataka iz baze	42
76. Brisanje selektovanog podataka	42
77. Povezivanje kombo boksa i baze	42
78. Ubacivanje vrednosti u tekst polja selekcijom iz kombo boksa.....	42
79. Ubacivanje vrednosti u tekst polja selekcijom iz list boksa	42
80. Prikaz podataka iz tabele u DataGridView	42
81. Povezivanje grafikona sa bazom podataka	42
82. Prikaz podatke iz dataGridView u TextBox-u	43
83. Prenos varijabli između formi (1)	43
84. Prenos varijabli između formi (2)	43

REŠENJA**Jednostavne forme (1- 12)**

1. Sabiranje dve celobrojne vrednosti iz programskog kôda	46
2. Sabiranje dve celobrojne vrednosti iz TextBox-a	47
3. Kalkulator sa četiri osnovne radnje	48
4. Računanje površine kruga	49
5. Računanje obima kruga	50
6. Računanje kvadratnog korena	51
7. Spajanje reči iz više TextBox-ova	52
8. Izračunavanje broja karaktera u nizu više reči	53
9. Konverzija slova	54

10. Konkatenacija	55
11. Konverzija broja i tekst	56
12. Konverzija teksta u broj	57

Konzolne aplikacije (13 – 21)

13. Aritmetička sredina	59
14. Lični podaci osobe	60
15. While petlja	61
16. While petlja i zbrajanje	62
17. For petlja i zbrajanje	63
18. For petlja i proizvod	64
19. Testiranje konzolnog prozora	65
20. Prosek unetih brojeva	66
21. Testiranje unetog broja	67
22. Izdvajanje najvećeg unetog broja	68
23. Upis ispitne ocene tekstom na osnovu broja	69
24. Switch petlja i obračun putarine na autoputu	70
25. Računanje prosečne brzine na autoputu	72

Složene forme (22 - 62)

26. Ispis poruke na ekranu	75
27. Otvaranje druge forme iz aktivne forme	76
28. Primena radio dugmeta i prijava ispita	77
29. Primena CheckBox i ankete	78
30. Dijalog poruke i odlučivanje	79
31. PictureBox i učitavanje slike	80
32. Prikazivanje sistemskog datuma i vremena	81
33. Pokretanje eksternih aplikacija	82
34. Kontrola Timer i štoperica	83
35. Validacija unosa u polje na formi	85
36. Konverzija dekadnog u binarni zapis	86
37. Konverzija dekadnog u oktalni zapis	87
38. Konverzija dekadnog u heksadekadni zapis	88
39. Menjanje slike u PictureBox-u iz liste	90
40. Kontrola Timer i odbrojavanje	91
41. Validacija unosa u polje na događaj KeyPress	93
42. Ograničenje unosa povodom broja znakova	95
43. Pravljenje tekst editora	96
44. Provera regularnosti unosa u predefinisana polja	97

45. Logovanje iz programskog kôda	99
46. ProgresBar – napredovanje	100
47. Šifriranje i dešifriranje unesenog teksta	101
48. Kontrola TreeView iz programskog kôda	103
49. Štampanje teksta iz polja	105
50. Kontrola dateTimePicker i razlika dva datuma	106
51. Držač mesta TextBox-a	108
52. Upotreba kontrole trackBar (1)	110
53. Upotreba kontrole trackBar (2)	111
54. Kontrola PictureBox i rotiranje slike	112
55. Klizni panel – otkrivanje i sakrivanje kontrole panel.	114
56. Datum isteka aplikacije	116
57. Padajući meni	117
58. Traka menija	119
59. Kontekstni meni	121
60. TabControl-e	123
61. Open-Save file dialog	125
62. Višenamenski interfejs	126

Rad sa bazama podataka (63 - 84)

(Projekat „Student”)

63. Provera konekcije	130
64. Punjenje dataGridView podacima iz tabele	131
65. Snimanje novog zapisa	133
66. Prenošnje fokusa iz kontrole u kontrolu tipkom Enter	135
67. Brisanje zapisa i pražnjenje kontrola za unos	136
68. Sinhronizacija kontrola za unos sa dataGridView kontrolom	138
69. Ažuriranje zapisa	139
70. Osvežavanje prikaza	141

(Projekat „Zaposleni”)

71. Provera konekcije i obrada grške	142
72. Pravljenje forme za logovanje	143
73. Dodavanje slika i ikona na formu i otvaranje druge forme.....	144
74. Insertovanje i snimanje podataka u bazu	145
75. Ažuriranje podataka iz baze	147
76. Brisanje selektovanog podataka	148
77. Povezivanje kombo boksa i baze	149
78. Ubacivanje vrednosti u tekst polja selekcijom iz kombo boksa.....	151

79. Ubacivanje vrednosti u tekst polja selekcijom iz list boksa	153
80. Prikaz podataka iz tabele u DataGridView	155
81. Povezivanje grafikona sa bazom podataka	156
82. Prikaz podatke iz dataGridView u TextBox-u	157
83. Prenos varijabli između formi (1)	158
84. Prenos varijabli između formi (2)	160

Napomena autora:

Ova zbirka nije pravljena s namerom da se pomoću nje može vršiti obuka u radu alatom *Microsoft Visual Studio Community 2017*, već se čitalac upućuje da te veštine savlada i stekne uz pomoć adekvatnog udžbenika ili tutorijala. Zbirka je namenjena svima koji žele da svoje teoretsko znanje iz programiranja provere, upotpune i nadograde korišćenjem *C# programskog jezika*.

Zadaci su sačinjeni tako da je akcenat na kodiranju. U odeljaku „*Uvodni zadaci*” su odmah data i rešenja, a namera je bila uputiti čitaoca ili podseti ga na sintaksu nekih jednostavnih programerskih zahteva. Odeljak „*Jednostavne forme*” je koncipiran tako da je fokus na par osnovnih kontrola koje se najčešće koriste pri izradi aplikacija. U odeljku „*Konzolne aplikacije*” su zastupljeni zadaci sa strukturama kave su petlje i iskazi odlučivanja, dok se u odeljku „*Složene forme*” pretežno nalaze zadaci sa upotrebom kontrolnih objekta iz šireg spektra alata *Microsoft Visual Studio Community 2017*. Na kraju, u odeljku „*Rad sa bazama podataka*”, nalaze se zadaci koji tretiraju upotrebu „*C# programskog jezika sa tabelama MS Access-a*”. U većini slučajeva, zadaci iz ovog odeljka su povezani tako da se rešenje pojedinog zadatka temelji na rešenju prethodnog.

Svi kontrolni objekti alata *Visual Studio* pisani su u zadacima onako kako su imenovani u ToolBox-u. Italikom rešenja, naravno, imaju specifičnu sintaksu koje se treba strogo pridržavati s obzirom na to da je *C# programski jezik* visoko senzitivnan. Stoga obratite pažnju na pisanje malih i velikih slova u programskom kôdu.

Komunikacija sa autorom na mail: mzahorjanski@raf.rs

Uputstvo

Za izradu zadataka neophodno je instalirati *Microsoft Visual Studio Community 2017*". Pri instalaciji odaberite opciju *.NET Desktop Development*" kako biste imali mogućnost da kasnije, prilikom otvaranja novih projekata u okviru jezika C#, birate *Windows Classic Desktop*" i podopciju *Windows Forms App (.NET Framework)*".

Za sva rešenja zadataka korišćene su standardne biblioteke koje se dobijaju izborom gore pomenutih opcija prilikom otvaranja novih projekata, pa čitalac ne bi trebalo da ima problema ni sa jednom naredbom prikazanom u programskom kôdu rešenja zadatka. Biblioteke koje će biti potrebno povremeno priključiti direktivom *using* u programskom kôdu su sastavni deo verzije pomenutog alata i u tekstu zadatka će biti naglašeno kada ih treba dodati, odnosno priključiti.

Uvodni zadaci

Otvorite jedan projekat za veћbanje. Na formi koju ste pritom dobili, isprobajte sve sledeće osobine i funkcionalnosti objekata iz palete alata, koristeći sledeće: Button, Label, TextBox, RichTextBox i PictureBox. Za ispitivanje koristite događaj forme Load. Iza kosih crta, nadesno, nalaze se komentari i sugestije, a ispod toga, fontom Courier New, nalazi se ono što treba upisati u Form.cs.

// **BUTTON**

// Kako da u programskom kôdu promenite naziv koji se vidi na
// dugmetu.

```
button1.Text = „Dugme“;
```

// Pomoću sledeće linije kôda u opštem obliku, poravnava se natpis na
// dugmetu.

```
(svojstvo TextAlign).Naredba
```

// Osobina ContentAlignment sluţi za odabir načina na koji će se sav
// tekst koji se pojavljuje na dugmetu poravnati. U ovom slučaju će
// se natpis poravnati u gornjem levom uglu („TopLeft”).

```
button1.TextAlign = ContentAlignment.TopLeft;
```

// Pomoću sledeće linije kôda, menja se boja dugmeta (svojstvo
// *BackColor*). Naredba Color sluţi za odabir boje, a u ovom slučaju
// smo odabrali žutu boju (Yellow).

```
button1.BackColor = Color.Yellow;
```

```
// Pomoću sledeće linije kôda, menja se izgled kursora (svojstvo  
// Cursor). Naredba Cursor služi za biranje izgleda kursora, a u  
// ovom slučaju smo izabrali krstasti izgled (Cross).
```

```
button1.Cursor = Cursors.Cross;
```

```
// Pomoću sledeće linije kôda, menja se izgled dugmeta (svojstvo  
// FlatStyle). Naredba FlatStyle služi za izbor stila, a u ovom  
// slučaju smo izabrali Flat stil.
```

```
button1.FlatStyle = FlatStyle.Flat;
```

```
// Pomoću sledeće linije kôda, menja se boja ivice dugmeta (svojstvo  
// FlatAppearance, podsvojstvo BorderColor). Naredba Color služi  
// za odabir boje, a u ovom slučaju smo odabrali zelenu boju  
// (Green).
```

```
button1.FlatAppearance.BorderColor = Color.Green;
```

```
// Sledećom linijom kôda menja se veličina ivice dugmeta  
// (svojstvo FlatAppearance, podsvojstvo BorderSize). U ovom  
// slučaju smo odabrali da velicina bude 3.
```

```
button1.FlatAppearance.BorderSize = 3;
```

```
// Pomoću sledeće linije kôda menja se boja dugmeta u trenutku kada  
// kursor miša stoji na dugmetu, a levi taster na mišu je pritisnut kako  
// bi kursor išao nadole.  
// (Svojstvo FlatAppearance, podsvojstvo MouseDownBackColor).  
// Naredba Color služi za odabir boje, a u  
// ovom slučaju smo odabrali boju („Violet”).
```

```
button1.FlatAppearance.MouseDownBackColor=  
Color.Violet;
```

// Pomoću sledeće linije kôda se menja boja dugmeta kada se pređe
// mišem preko njega (svojstvo FlatAppearance, podsvojstvo
// MouseOverBackColor). Naredba Color služi za odabir boje, a u
// ovom slučaju smo odabrali ljubicastu boju (Purple).

```
button1.FlatAppearance.MouseOverBackColor=  
Color.Purple;
```

// Pomoću sledeće linije kôda se menja visina dugmeta (podsvojstvo
// Height). U ovom slučaju smo postavili da visina bude 100.

```
button1.Height = 100;
```

// Pomoću sledeće linije kôda se menja širina dugmeta (podsvojstvo
// Width). U ovom slučaju smo postavili da širina bude 150.

```
button1.Width = 150;
```

// Pomoću sledeće linije kôda se menja redosled po kome će dugme
// da dobije fokus (svojstvo TabIndex). U ovom slučaju je
// postavljeno na 1.

```
button1.TabIndex = 1;
```

// Pomoću sledeće linije kôda se utvrđuje da li se tasterom Tab
// može dugme postaviti u žižu (svojstvo TabStop). U ovom slučaju
// može, jer je postavljeno na vrednost True. Da je postavljeno na
// vrednost „False”, ne bi moglo.

```
button1.TabStop = true;
```

// Pomoću sledeće linije koda se određuje da li će dugme da bude
// vidljivo ili sakriveno kada se pokrene program (svojstvo
// Visible).

```
button1.Visible = true;
```

// LABELA

// Postavljanje teksta u Labelu.

```
label1.Text = „Ovo je labela!“;
```

// Na ovaj način će se dimenzije Labele prilagoditi tekstu koji
// je u njoj ispisan.

```
label1.AutoSize = true;
```

// Promena boje pozadine Labele.

```
label1.BackColor = Color.Cyan;
```

// Na ovaj način, odnosno kada se mišem pređe preko Labele, kursor
// će poprimiti drugi izgled.

```
label1.Cursor = Cursors.Hand;
```

// Promena izgleda ivica Labele.

```
label1.BorderStyle = BorderStyle.Fixed3D;
```

// Labela će biti vidljiva kada se pokrene program.

```
label1.Visible = true;
```

// TEXTBOX

// Pošto će najčešće biti potrebno da TextBox bude prazan,
// onda ćemo pomoću ove linije koda to i postići. Između
// znakova navodnika ne pišemo ništa.

```
textBox1.Text = „“;
```

// Promena boje pozadine TextBox-a.

```
textBox1.BackColor = Color.YellowGreen;
```

// Promena izgleda ivica.

```
textBox1.BorderStyle = BorderStyle.Fixed3D;
```

// Promena izgleda kusora.

```
textBox1.Cursor = Cursors.HSplit;
```

// Sledeća sintaksa omogućava unos teksta u više redova.

```
textBox1.Multiline = true;
```

// Postavlja strelice za pomeranje teksta gore dole. Ovo
// ima smisla stavljati samo kada je svojstvo Multiline
// postavljeno na True.

```
textBox1.ScrollBars = ScrollBars.Vertical;
```

// Zadavanje visine TextBox-a ima smisla
// samo kada je svojstvo Multiline postavljeno na True.

```
textBox1.Height = 50;
```

// Zadavanje širine TextBox-a.

```
textBox1.Width = 150;
```

// Zadavanje redosleda po kome će TextBox biti u žiži.

```
textBox1.TabIndex = 1;
```

// Omogućava da se pomoću tastera Tab žiža premesti na ovaj
// objekat.

```
textBox1.TabStop = true;
```

// TextBox će biti vidljiv kada se pokrene program.

```
textBox1.Visible = true;
```

// RICHTEXTBOX

// Pošto će najčešće biti potrebno da RichTextBox bude prazan,
// onda ćemo pomoću ove linije koda to i postići. Između
// znakova navodnika ništa ne pišemo.

```
richTextBox1.Text = "";
```

// Promena boje pozadine RichTextBox-a.

```
richTextBox1.BackColor = Color.Thistle;
```

// Promena izgleda ivica.

```
richTextBox1.BorderStyle = BorderStyle.Fixed3D;
```

// Promena izgleda kursora.

```
richTextBox1.Cursor = Cursors.IBeam;
```

// Detekcija linkova.

```
richTextBox1.DetectUrls = true;
```

// Promena boje teksta koji će biti ispisan u RichTextBox-u.

```
richTextBox1.ForeColor = Color.Magenta;
```

// Omogućava unos teksta u više redova.

```
richTextBox1.Multiline = true;
```

// Zadavanje visine.

```
richTextBox1.Height = 50;
```

// Zadavanje širine.

```
richTextBox1.Width = 150;
```

// Zadavanje redosleda po kome će RichTextBox da bude u žiži.

```
richTextBox1.TabIndex = 1;
```

// Omogućava da se pomoću Tab tastera žiža premesti na ovaj objekat.

```
richTextBox1.TabStop = true;
```

// RichTextBox će biti vidljiv kada se pokrene program.

```
richTextBox1.Visible = true;
```

// PICTUREBOX

// Promena boje pozadine PictureBox-a.

```
pictureBox1.BackColor = Color.BlueViolet;
```

// Promena izgleda ivica.

```
pictureBox1.BorderStyle = BorderStyle.FixedSingle;
```

// Promena izgleda kursora.

```
pictureBox1.Cursor = Cursors.Arrow;
```

// Zadavanje visine PictureBox-a.

```
pictureBox1.Height = 180;
```

// Zadavanje širine PictureBox-a.

```
pictureBox1.Width = 200;
```

// PictureBox će biti vidljiv kada se pokrene program.

```
pictureBox1.Visible = true;
```


// TOOLTIP

```
// Na formi mora da bude aktiviran događaj Form_Load.
// Treba dodati i kontrolu tooltip, a na kraju u kôdu dodati i sledeću
// liniju:

    tooltip1.SetToolTip(button1, „Pritisnite dugme“);

// Na ovaj način se u Label1 ispisuje prvo ono što piše u prvom
// TextBox-u, pa onda pored toga ono što piše u drugom TextBox-u.
// Međutim, između imena i prezimena treba da stoji razmak, pa
// zato i dodajemo prazno mesto između znakova navodnika. Znak
// plus je neophodan jer se pomoću njega povezuju ove naredbe.

    label1.Text = textBox3.Text + „ „ + textBox2.Text;

// Na ovaj način se u TextBox1 ispisuje ovaj isti sadržaj.

    textBox1.Text = textBox3.Text + „ „ + textBox2.Text;

// Na ovaj način se u MessageBox ispisuje ovaj isti sadržaj.

    MessageBox.Show(textBox3.Text + „ „ + textBox2.Text);

// Na ovaj način se u RichTextBox1 ispisuje ovaj isti sadržaj.

    richTextBox1.Text = textBox3.Text + „ „ + textBox2.Text;

// Na ovaj način se iz tekuće forme klikom na dugme pokreće Form2.

private void button8_Click(object sender, EventArgs e)
{
    var form = new Form2();
    form.Show();
}
```

// Na ovaj naćin se postavlja fokus na Źeljenu kontrolu nakon
// otvaranja forme.

```
private void Form1_Load(object sender, EventArgs e)
{
    this.ActiveControl = textBox1;    // prvi naćin
    textBox1.Select();                // drugi naćin
}
```

// Na sledeći naćin se prelazi sa kontrole na sledeću kontrolu
// pritiskom na taster Enter.

```
private void Model_KeyDown(object sender, KeyEventArgs e)
{
    if (e.KeyCode == Keys.Enter)
    {
        SendKeys.Send(„{TAB}“);    // prvi naćin
        // ako nije susedna onda
        // imeKontrole.Focus();    // drugi naćin
    }
}
```