



Природно-математички факултет  
Универзитет у Бањој Луци

# Основни концепти C++ програмирања – средње школе

*Аутори: Драган Матић, Милана Грбић,  
Милан Предојевић, Ненад Вилендечић*

2025

# Садржај

<b>Увод</b>	<b>3</b>
<b>Задаци</b>	<b>4</b>
1   Задатак: Такси . . . . .	5
2   Задатак: Уметање броја у број . . . . .	6
3   Задатак: ASCII . . . . .	8
<b>Домаћи</b>	<b>10</b>
1   Задатак: Издвајање новчаница . . . . .	10
2   Задатак: Збирови по три странице правоугаоника . . . . .	10
3   Задатак: Шах . . . . .	10
4   Задатак: Исправан датум . . . . .	10

## Увод

На првом предавању Прољетне школе програмирања за ученике средњих школа обрађене су основне теме из програмирања, укључујући промјенљиве, типове података, операторе и наредбу гранања if - else.

У наставку се налазе задаци који су рађени током предавања, са циљем да ученици примјене и утврде стечено знање кроз практичне примјере. На крају документа налазе се задаци за домаћи рад, који служе за додатно вјежбање и самостално истраживање обрађених концепата.

# Задаци

## 1. Задатак: Такси

У такси су истовремено ушла три путника, али су излазили један по један. Договорили су се да у плаћању сваког дијела вожње подједнако учествују путници који су у том дијелу вожње били у таксију. Ако се зна стање на таксиметру у КМ када је изашао свако од њих, колико треба да плати путник који је изашао први, колико други, а колико трећи?

### Примјер извршавања

```
Unesite stanje na taksimetru kada je izasao prvi putnik: 12
Unesite stanje na taksimetru kada je izasao drugi putnik: 15
Unesite stanje na taksimetru kada je izasao treci putnik: 20
Prvi putnik treba da plati: 4 KM
Drugi putnik treba da plati: 5.5 KM
Treci putnik treba da plati: 10.5 KM
```

### Рјешење

```
1     #include <iostream>
2
3     using namespace std;
4
5     int main() {
6         double p1, p2, p3;
7         cout << "Unesite stanje na taksimetru kada je izasao prvi
8 putnik: ";
9         cin >> p1;
10        cout << "Unesite stanje na taksimetru kada je izasao
11 drugi putnik: ";
12        cin >> p2;
13        cout << "Unesite stanje na taksimetru kada je izasao
14 treci putnik: ";
15        cin >> p3;
16
17        double prvi = p1 / 3;
18        double drugi = (p2 - prvi) / 2;
19        double treci = p3 - prvi - drugi;
20
21        cout << "Prvi putnik treba da plati: " << prvi << " KM"
22 << endl;
23        cout << "Drugi putnik treba da plati: " << drugi << " KM"
24 << endl;
25        cout << "Treci putnik treba da plati: " << treci << " KM"
26 << endl;
27
28        return 0;
29    }
```

## 2. Задатак: Уметање броја у број

Написати програм који учитава позитиван цијели број  $n$  и позитиван двоцифрен број  $m$  и исписује број добијен уметањем броја  $m$  између цифре стотине и хиљаде броја  $n$ .

### Примјер извршавања

Unesite pozitivan broj n (broj sa najmanje 4 cifre): 12345

Unesite pozitivan dvocifren broj m: 67

Rezultat: 1267345

### Рјешење

```
1     #include <iostream>
2
3     using namespace std;
4
5     int main() {
6         // Deklaracija promjenljivih
7         int n, m;
8
9         // Unos brojeva
10        cout << "Unesite pozitivan broj n (broj sa najmanje 4
11 cifre): ";
12        cin >> n;
13
14        cout << "Unesite pozitivan dvocifren broj m: ";
15        cin >> m;
16
17        // Provjera da li je i m dvocifren broj
18        if (m < 10 || m > 99) {
19            cout << "Uneseni broj m nije dvocifren." << endl;
20            return 1;
21        }
22
23        // Uzimanje cifara broja n
24        int cioDio = n / 1000;
25        int ostatak = n % 1000;
26
27        int noviBroj = cioDio*100000 + m*1000 + ostatak;
28
29        cout<<"Novi broj je: "<<noviBroj<<endl;
30
31
32        return 0;
33    }
34
```

## Питање

Зашто имамо нежељен испис испод ако су  $m$  и  $n$  типа *int*?

Unesite pozitivan broj n (broj sa najmanje 4 cifre): 50000000

Unesite pozitivan dvocifren broj m: 12

Rezultat: 705044704

### 3. Задатак: ASCII

Написати програм који за уčitани карактер испишује унесени карактер и његов ASCII код. Уколико је унесени карактер мало (велико) слово, исписати и одговарајуће велико (мало) слово и његов ASCII код.

#### Примјер извршавања 1

Unesite karakter: b  
Uneseni karakter: b, ASCII kod: 98  
Odgovarajuce veliko slovo: B, ASCII kod: 66

#### Примјер извршавања 2

Unesite karakter: 5  
Uneseni karakter: 5, ASCII kod: 53

#### Рјешење

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     // Deklaracija promjenljive za karakter
7     char ch;
8
9     // Unos karaktera
10    cout << "Unesite karakter: ";
11    cin >> ch;
12
13    // Ispis unesenog karaktera i njegovog ASCII koda
14    cout << "Uneseni karakter: " << ch << ", ASCII kod: " <<
15    int(ch) << endl;
16
17    // Ako je uneseni karakter malo slovo, ispisujemo
18    // odgovarajuce veliko slovo
19    if (ch >= 'a' && ch <= 'z') {
20        char veliko = ch - 'a' + 'A'; // Pretvaranje u veliko
21        slovo
22        cout << "Odgovarajuce veliko slovo: " << veliko << ",
23        ASCII kod: " << int(veliko) << endl;
24    }
25
26    // Ako je uneseni karakter veliko slovo, ispisujemo
27    // odgovarajuce malo slovo
28    else if (ch >= 'A' && ch <= 'Z') {
29        char malo = ch - 'A' + 'a'; // Pretvaranje u malo slovo
30        cout << "Odgovarajuce malo slovo: " << malo << ", ASCII
31        kod: " << int(malo) << endl;
32    }
33 }
```



27

28

29

```
return 0;
```

```
}
```

# Домаћи рад

## 1. Задатак: Издајање новчаница

Написати програм који за унесу цијену производа исписује најмањи број новчаница које је потребно издвојити приликом плаћања производа. На располагању су новчанице од 100, 50, 20, 10, 5, 2 и 1 КМ.

### Примјер извршавања

```
Unesite cijenu proizvoda:8367
8367 = 83*100 + 1*50 + 0*20 + 1*10 + 1*5 + 1*2 + 0*1
Најмањи број новчаница је: 87.
```

## 2. Задатак: Збирови по три странице правоугаоника

Ако је збир неке три странице правоугаоника  $m$ , а збир неке друге три странице истог правоугаоника  $n$ , написати програм којим се одређује обим и површина тог правоугаоника.

### Примјер извршавања

```
Unesite prvi zbir tri stranice: 8
Unesite drugi zbir tri stranice: 10
Stranica a: 2
Stranica b: 4
Obim: 12
Povrsina: 8
```

## 3. Задатак: Шах

Поља шаховске табле се дефинише паром цијелих бројева  $(x, y)$ ,  $1 \leq x, y \leq 8$ , гдје је  $x$  редни број реда, а  $y$  редни број колоне. Написати програм који за унесене парове  $(k, l)$  и  $(m, n)$  провјерава

- да ли су поља  $(k, l)$  и  $(m, n)$  исте боје,
- да ли краљица са  $(k, l)$  угрожава поље  $(m, n)$ ,
- да ли коњ са  $(k, l)$  угрожава поље  $(m, n)$ .

Претпоставити да је поље  $(1, 1)$  црно и да представља доњи лијеви угао шаховске табле. У случају неисправног уноса, исписати одговарајућу поруку о грешци.

## 4. Задатак: Исправан датум

Написати програм који за унети датум у формату дан.месец.година. провјерава да ли је коректан.

### **Примјер извршавања 1**

Unesite datum: 25.11.1983.  
Datum je korektan.

### **Примјер извршавања 2**

Unesite datum: 1.17.2004.  
Datum nije korektan.