
Tema 2

TEME

- Aritmetičke operacije
- Kastrovanje tipova
- Matematicka biblioteka
- Koristenje konstante Pi
- Imenovane konstante
- Funkcije
 - Povratna vrednost funkcije
 - void
 - Prenos parametara po vrednosti i po referenci

PRIMER 1: Aritmetičke operacije / ZADRŽATI SE NA OVOM PRIMERU SAMO AKO UCENIKU NISU POZNATE OVE OPERACIJE IZ DRUGIH PROGRAMSKIH JEZIKA!

```
#include <iostream>
using namespace std;

int main() {
    cout << "Unesi integere" << endl;
    int a, b, c;
    cin >> a >> b ;
    cout << a+b << endl;
    cout << a-b << endl;
    cout << a*b << endl;
    cout << a%b << endl;    // ostatak
    cout << a/b << endl;
    cout << a/(double)b << endl;
    return 0;
}
```

OBJAŠNENJE:

- **Primetimo da nema operatora // kao u Pythonu jer je to ovde komentar**
- Realno deljenje se postize cast-om delioca na realan broj
- Prioriteti operacija su kao u matematici

PRIMER 2: Matematicke funkcije // OBAVEZAN PRIMER

```
#include <iostream>
#include <cmath>
using namespace std;
```

```

int main() {
    cout << "Unesi integere i float" << endl;
    int a, b;
    float c;
    cin >> a >> b >> c;
    cout << pow(a, b) << endl;
    cout << sqrt(a) << endl;
    cout << ceil(c) << endl;
    cout << floor(c) << endl;
    cout << remainder(a, b) << endl;
    cout << round(c) << endl;
    return 0;
}

```

OBJAŠNENJE:

- Primetimo da su funkcije kao u Pythonu
- Round radi drugacije nego u Pythonu (nema fore za parnost za 1.5 i 2.5, ovde se uvek zaokružuje na vise)
- Dodatno ima jos mnogo math trigonometrijskih i logaritamskih funkcija

PRIMER 3: PI broj - za unetu površinu kruga izracunati obim sa 2 decimalne // OBAVEZAN PRIMER

```

#define _USE_MATH_DEFINES
#include <iostream>
#include <iomanip>
#include <cmath>

using namespace std;
int main() {
    double P, r, O;
    cin >> P; // učitavamo površinu
    r = sqrt(P / M_PI); // izračunavamo poluprečnik
    O = 2 * r * M_PI; // izračunavamo obim
    cout << setprecision(2) << O << endl;
}

```

OBJAŠNENJE:

- Pi nije standardizovana kao konstanta u jeziku direktno, ali jeste ukoliko se koristi makro `_USE_MATH_DEFINES`, ako se on navede pre math biblioteke, tada se može u programu koristiti `M_PI`
- `Setprecision` se nalazi u `iomanip`, označava broj decimala koje hoćemo da prikazemo
- Moguće je koristiti i kvalifikatore operatora `cout`: `fixed` i `showpoint` za prikaz decimalnih brojeva
- Drugi način za korišćenje PI konstante je da je sami definisemo `const double PI = 3.14159265;`

- Ovo const oznacava da je ta promenljiva fiksirana svojom vrednoscu

PRIMER 4: Napraviti funkcije za racunanje obima i povrsine jednakostranicnog pravougaonika // OBAVEZNO URADITI KAO I DODATAK SA REFERENCOM ILI PRETHODNI PRIMER URADITI SA FUNKCIJOM I DODATKOM ZA PROSLLEDJIVANJE PO REFERENCI!

```
#include <iostream>
#include <cmath>

using namespace std;

void naslov() {
    cout << "Ovo je nas matematicki pametni digitron" << endl;
    cout << "Funkcionalnosti su: " << endl;
    cout << "* obim jednakostranicnog trougla" << endl;
    cout << "* povrsina jednakostranicnog trougla" << endl;
}

double obim(double stranica) {
    return 3*stranica;
}

double povrsina(double stranica) {
    double h = stranica * sqrt(3) / 2.0;
    double p = stranica * h / 2.0;
    return p;
}

int main() {
    naslov()
    cout << "Unesi stranicu " << endl;
    double a;
    cin >> a;
    double O = obim(a);
    double P = povrsina(a);
    cout << "Obim: " << O << endl;
    cout << "Povrsina: " << P << endl;
}
```

OBJAŠNENJE:

- Definisane funkcije, sintaksa i povratni tip
- Poziv funkcija i prosledjivanje argumenata funkciji
- Funkcije bez returna - void
- Matematicke funkcije i aritmetičke operacije, upotreba kroz primer
- Pomeriti liniju using namespace std nize, odmah iznad maina - greska ce biti da se cout gore u funkciji ne moze kompajlirati

- Funkcije sa vise povratnih vrednosti - to se postize ako se parametri funkciji prosledjuju po referenci (&)
 - Dodati funkciju izracunajSve
 - Dodati deo u mainu kojim se dodaje deo za testiranje ove fuunkcije
- Referenca je adresa u memoriji
- Parametri funkcija koji su "preneseni" po referenci su ulazno izlazni podaci neke funkcije

```
void izracunajSve(double stranica, double& pov, double& ob) {
    pov = povrsina(stranica);
    ob = obim(stranica);
}
```

... // na kraju maina:

```
double o_1, p_1;
izracunajSve(a, p_1, o_1);
cout << "Obim: " << o_1 << endl;
cout << "Povrsina: " << p_1 << endl;
}
```

BONUS PRIMERI - NE MORAJU DA SE RADE AKO NEMA VREMENA, ZADATI ZA VEZBU

PRIMER 5: Povrsina trougla ako su data temena u 2d prostoru

IDEJA:

- Na osnovu temena izracunati duzine stranica (analiticka geometrija)
- Na osnovu duzine stranica, izracunati s
- Na osnovu stranica i s, izracunati povrsinu preko Heronovog obrasca

```
#define _USE_MATH_DEFINES
#include <iostream>
#include <iomanip>
#include <cmath>

using namespace std;

// rastojanje izmedju tacaka (x1, y1) i (x2, y2)
double rastojanje(double x1, double y1, double x2, double y2) {
    double dx = x2 - x1;
    double dy = y2 - y1;
    return sqrt(dx*dx + dy*dy);
}

// povrsina trougla za zadate duzine stranica
```

```

double površinaTrougla(double a, double b, double c) {
// Heronov obrazac
    double s = (a + b + c) / 2;
    return sqrt(s*(s-a)*(s-b)*(s-c));
}

int main() {
    // koordinate tacaka trougla
    double Ax, Ay, Bx, By, Cx, Cy;
    cin >> Ax >> Ay >> Bx >> By >> Cx >> Cy;
    // duzine stranica trougla
    double a = rastojanje(Bx, By, Cx, Cy);
    double b = rastojanje(Ax, Ay, Cx, Cy);
    double c = rastojanje(Ax, Ay, Bx, By);
    // površina trougla
    double P = površinaTrougla(a, b, c);
    cout << setprecision(5) << P << endl;
    return 0;
}

```

OBJAŠNENJE:

- Sta bi ovde moglo da se prosledi po referenci
- Ovakve stvari mogu biti takmicarske - u principu postoji neki matematički dio koji je potrebno prvo resiti pa tek onda isprogramirati

PRIMER 6: Prosek na takmicenju (zadatak sa takmicenja)

Imamo N ucesnika na takmicenju iz matematike. Izvestaj o broju poena je stampan na 2 strane, na dnu prve je prosek za tu stranu p1, a na drugoj strani ima K ucenika i njihov prosek p2.

Potrebno je napisati program kojim se racuna prosek svih ucenika, a ulaz je broj N, broj K, broj p1, broj p2.

Primer ulaza:

80 30 78.20 89.30

Izlaz je 82.36

MATEMATICKO RESENJE:

Stranica1 = 80-30 = 50

Prosek1: suma1 / 50 = 78.20 -> suma1 = 50 * 78.20

Prosek2: suma2 / 30 = 89.30 -> suma2 = 30 * 89.30

glavniProsek: (suma1 + suma2) / 80

RESENJE:

```
#define _USE_MATH_DEFINES
#include <iostream>
#include <iomanip>
#include <cmath>

using namespace std;

int main() {
    int N, K;
    double p1, p2, glavniProsek;
    double s1, s2;
    cin >> N >> K >> p1 >> p2;
    s1 = (N-K) * p1;
    s2 = K * p2;
    cout << s1 + s2 << endl;
    glavniProsek = (s1+s2) / (double) N;
    cout << glavniProsek << endl;
    return 0;
}
```

OBJAŠNENJE:

- Ulaz može da se ubacuje i sa spaceom i sa enterom između
- Izlazu može da se doda `setprecision(2)` ali nešto pravi problem