

\*\*\*\*\*

# Tema 4

\*\*\*\*\*

## TEME

- Naredbe ponavljanja
  - For
  - While
  - Do while
- Break i continue
  - Break prekid cele naredbe ponavljanja i izlazak iz ciklusa
  - Continue prekid tekuće iteracije
  - Ako se koriste u ugnjezdenim petljama ponavljanja, odnose se samo na onu koja im je bliza
- Sintaksa - objasniti razlike u odnosu na Python
  - while (izraz)  
naredba
  - for (izraz1, izraz2, izraz3) // izraz1 i izraz3 su dodele ili inkrementacije, a izraz2  
relacija  
naredba
  - do {  
naredbe  
} while (uslov)

## PRIMER 1: Ispisati cifre visecifrenog broja sa desna na levo

### KORACI RESAVANJA:

- Objasniti da desnu cifru mozemo dobiti kada radimo % 10 operaciju
- Narednu cifru mozemo dobiti na isti nacin ako bismo nekako uklonili tu poslenju cifru, a nju mozemo ukloniti ako uradimo /10 operaciju u integeru
- Identifikovati da se to onda ponavlja dok god imamo cifara, tj dok je broj veci od nule
- Setovati promenljivu i njen unos
- Konstruisati do while petlju sa potrebnim uslovom
- Definisati sta se ponavlja i implementirati
- Testirati

### FINALNA IMPLEMENTACIJA:

```
#include<stdio.h>
int main(){
    int n;
    printf("Unesi visecifren broj\n");
    scanf("%d", &n);
```

```
do {
    printf("%d ", n%10);
    n /= 10;
} while(n>0);
return 0;
}
```

## OBJASNJENJE

- Objasniti kako bi u Pythonu ovo uradili
- Da li moze i sa while obicnom petljom da se uradi? Probati.

**PRIMER 2: Ispisati sve brojeve od 1 do 100 ali bez onih koji su deljivi sa 5. Demonstrirati korištenje naredbe continue (preskakanje samo tekuće iteracije).**

## KORACI RESAVANJA:

- Objasniti algoritam i da nam treba nesto sto ce da prolazi kroz sve brojeve od 1 do 100
- Potrebno je koristisi for naredbu i if
- Napraviti prvo obican if bez continue naredbe, sa uslovom da je  $i\%5 \neq 0$
- Testirati
- Napraviti if da bude sa continue naredbom
- Testirati

## FINALNA IMPLEMENTACIJA:

```
#include<stdio.h>
int main() {
    for (int i = 1; i <= 100; i++) {
        if (i % 5 == 0) continue;
        printf("%d ", i);
    }
    return 0;
}
```

## OBJASNJENJE

- U printu mora biti i space da bi se brojevi odvojeno ispisivali
- Probati da bude prefiksni inkrement, sta to menja, koje su onda vrednosti za i?

**PRIMER 3: Suma brojeva do X**

## KORACI RESAVANJA:

- Objasniti algoritam i da sada opseg u for petlji ide do unetog broja
- Da li je poznat ovaj algoritam iz Pythona

- Koja je matematička formula za zbir prvih x brojeva
- Testirati

### FINALNA IMPLEMENTACIJA:

```
#include <stdio.h>
int main()
{
    int num, count, sum = 0;

    printf("Unesi pozitivan broj: ");
    scanf("%d", &num);

    for(count = 1; count <= num; ++count)
    {
        sum += count;
    }

    printf("Sum = %d", sum);

    return 0;
}
```

### OBJASNJENJE

- Ovaj zadatak može učenik sam da uradi nakon diskusije

### PRIMER 4: Sta radi naredni kod

```
#include<stdio.h>
int main(){
    int a = 10;
    for (;;) {
        printf("%d ", a);
    }
    return 0;
}
```

### OBJASNJENJE

- For može da bude beskonačan!

**PRIMER 5: Ispisati sve kvadrate parnih brojeva počev od broja 2, dok rezultat ne postane trocifren.**

### KORACI RESAVANJA:

- Prodiskutovati sta su ulazni podaci, koju petlju koristiti, koji uslov postaviti
- Testirati

### FINALNA IMPLEMENTACIJA:

```
#include<stdio.h>
int main(){
    int a = 2;
    do {
        printf("%d ", a*a);
        a += 2;
    } while (a*a < 100);
    return 0;
}
```

### OBJASNJENJE

- Kako bi bilo uradjeno kroz neku drugu petlju?

### PRIMER 6: Za uneti broj sa terminala ispisati trougao od zvezdica.

#### Npr za broj 3

```
*
**
***
```

#### Npr za broj 5

```
*
**
***
****
*****
```

### KORACI RESAVANJA:

- Objasniti sta se ovde sve ponavlja, i unutar reda a i sami redovi
- Imacemo ugnjezdene naredbe ponavljanja
- Kako red x zna koliko treba da ima zvezdica?
- Napraviti dve petlje po razlicitoj promenljivoj
- Objasniti do koje vrednosti idu obe for petlje
- Gde se ispisuje enter
- Testirati

### FINALNA IMPLEMENTACIJA:

```
#include<stdio.h>
int main(){
```

```
int n;
printf("Unesi broj\n");
scanf("%d", &n);

for (int i = 1; i <= n; i++) {
    for (int j = 1; j <= i; j++){
        printf("*");
    }
    printf("\n");
}
return 0;
}
```

## **OBJASNJENJE**

- Ovaj primer ucenik moze da radi sam