

Programiranje u fizici

10. Naredbe iteracije

Prirodno-matematički fakultet u Nišu
Departman za fiziku

2. Konstante u programskom jeziku C
3. Operatori u programskom jeziku C
4. Algoritamsko resavanje problema
5. Vrste programskih naredbi
6. Naredbe iteracije

Uvod

for-petlja
while-petlja
do-while-petlja

Iteracije omogućavaju višestruko ponavljanje jednog dela programa. U programskom jeziku C postoje tri vrste iterativnih naredbi:

- for-petlja
- while-petlja
- do while-petlja

2. Konstante u programskom jeziku C
3. Operatori u programskom jeziku C
4. Algoritamsko resavanje problema
5. Vrste programskih naredbi
6. Naredbe iteracije

Uvod
for-petlja
while-petlja
do-while-petlja

Opšti oblik for-petlje je dat sa:

```
for (izraz1; izraz2; izraz3)  
    naredba petlje;
```

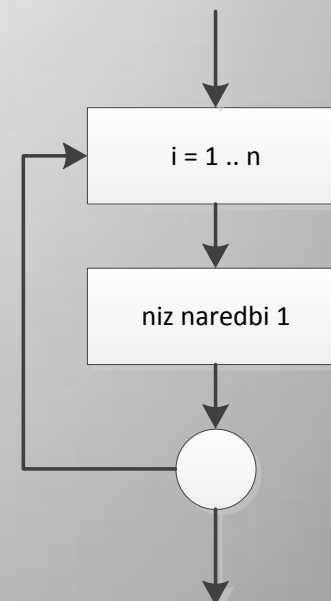
izraz1 inicijalizira promenljive, koje se pojavljuju u petlji

izraz2 predstavlja uslov, na osnovu kojeg se petlja izvršava ili prekida

izraz3 vrši ponovno dodeljivanje vrednosti promenljivima u petlji

naredba petlje može biti elementarna programska naredba ili blok naredbi.

Bilo koji od tri izraza u zagradi može biti izostavljen, ali se oba znaka ; moraju pisati.



2. Konstante u programskom jeziku C
3. Operatori u programskom jeziku C
4. Algoritamsko resavanje problema
5. Vrste programskih naredbi
6. Naredbe iteracije

Uvod
for-petlja
while-petlja
do-while-petlja

Primer: Napisati program za izračunavanje zbira aritmetičkog niza, tj. zbira prvih n prirodnih brojeva.

```
#include <stdio.h>
#include <stdlib.h>

int main()
{
    int zbir, ukup_br, i;

    printf ("Koliko prvih prirodnih brojeva treba sabrati? \n");
    scanf ("%d", &ukup_br);

    zbir = 0;
    for (i=1; i <= ukup_br; i++){
        zbir = zbir + i;

        printf("i= %d zbir= %d \n", i, zbir);
    }

    printf("Zbir prvih %d prirodnih brojeva je %d", ukup_br, zbir);

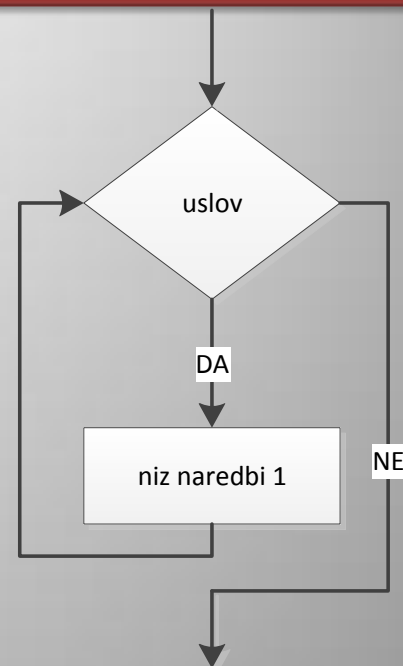
    return 0;
}
```

```
Koliko prvih prirodnih brojeva treba sabrati?
6
i= 1 zbir= 1
i= 2 zbir= 3
i= 3 zbir= 6
i= 4 zbir= 10
i= 5 zbir= 15
i= 6 zbir= 21
Zbir prvih 6 prirodnih brojeva je 21
```

2. Konstante u programskom jeziku C
3. Operatori u programskom jeziku C
4. Algoritamsko resavanje problema
5. Vrste programskih naredbi
6. Naredbe iteracije

Uvod
for-petlja
while-petlja
do-while-petlja

```
while (izraz)  
    naredba petlje;
```



To znači da **izraz** u while-petlji **mora** da promeni svoju logičku vrednost tokom ponavljanja petlje, da bi uopšte moglo doći do napuštanja petlje.

Ukoliko je **izraz** lažan, tj. ukoliko je njegova brojna vrednost jednaka nuli, naredba petlje se neće izvršiti i program će nastaviti rad sa prvom naredbom koja sledi while-petlju.

2. Konstante u programskom jeziku C
3. Operatori u programskom jeziku C
4. Algoritamsko resavanje problema
5. Vrste programskih naredbi
6. Naredbe iteracije

Uvod
for-petlja
while-petlja
do-while-petlja

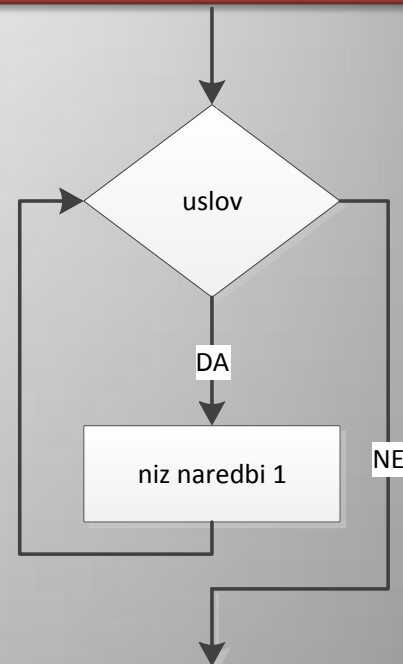
Opšti oblik while-petlje je dat sa:

```
while (izraz)  
    naredba petlje;
```

izraz je neka logička relacija ili promenljiva,
naredba petlje može biti elementarna programska naredba ili blok naredbi.

U while-petlji se prvo izračunava vrednost izraza u zagradi. Ukoliko je **izraz** logički tačan, tj. ukoliko je njegova brojna vrednost različita od nule, izvršiće se naredba petlje.

Nakon izvršenja ceo postupak će se ponoviti izračunavanjem izraz-a, sve dok on ne dobije logičku vrednost **lažan**.



2. Konstante u programskom jeziku C
3. Operatori u programskom jeziku C
4. Algoritamsko resavanje problema
5. Vrste programskih naredbi
6. Naredbe iteracije

Uvod
for-petlja
while-petlja
do-while-petlja

Primer: Napisati program za izračunavanje zbira aritmetičkog niza, tj. zbira prvih n prirodnih brojeva.

```
#include <stdio.h>
#include <stdlib.h>

int main()
{
    int zbir, ukup_br, i;

    printf ("Koliko prvih prirodnih brojeva treba sabrati? \n");
    scanf ("%d", &ukup_br);

    zbir = 0;
    i = 1;
    while (i <= ukup_br) {
        zbir = zbir + i;

        printf("i= %d zbir= %d \n", i, zbir);

        i++;
    }

    printf("Zbir prvih %d prirodnih brojeva je %d", ukup_br, zbir);

    return 0;
}
```

```
Koliko prvih prirodnih brojeva treba sabrati?
6
i= 1 zbir= 1
i= 2 zbir= 3
i= 3 zbir= 6
i= 4 zbir= 10
i= 5 zbir= 15
i= 6 zbir= 21
Zbir prvih 6 prirodnih brojeva je 21
```

2. Konstante u programskom jeziku C
3. Operatori u programskom jeziku C
4. Algoritamsko resavanje problema
5. Vrste programskih naredbi
6. Naredbe iteracije

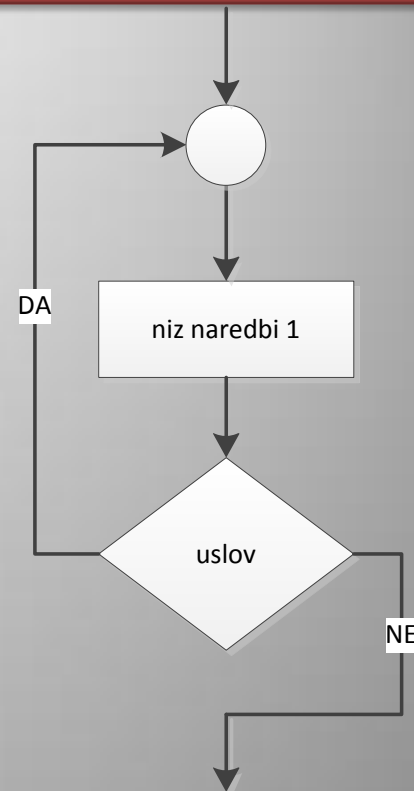
Uvod
for-petlja
while-petlja
do-while-petlja

Za razliku od **while** i **for-petlje**, zadovoljavanje datog uslova kod **do-while** petlje se ispituje tek na njenom **kraju**. To znači da će svaka **do-while**-petlja biti izvršena **najmanje jedanput**, što nije slučaj kod **while** i **for-petlje**.

do-while-petlja ima sledeći oblik:

```
do  
    naredba petlje  
while (izraz);
```

do while-petlja se izvršava sve dok je izraz u zagradi **tačan**, tj. dok je njegova brojna vrednost različita od nule.



2. Konstante u programskom jeziku C
3. Operatori u programskom jeziku C
4. Algoritamsko resavanje problema
5. Vrste programskih naredbi
6. Naredbe iteracije

Uvod
for-petlja
while-petlja
do-while-petlja

Primer: Napisati program za izračunavanje zbira aritmetičkog niza, tj. zbira prvih n prirodnih brojeva.

```
#include <stdio.h>
#include <stdlib.h>

int main()
{
    int zbir, ukup_br, i;

    printf ("Koliko prvih prirodnih brojeva treba sabrati? \n");
    scanf ("%d", &ukup_br);

    zbir = 0;
    i = 1;
    do {
        zbir = zbir + i;

        printf("i= %d zbir= %d \n", i, zbir);

        i++;
    }
    while (i <= ukup_br);

    printf("Zbir prvih %d prirodnih brojeva je %d", ukup_br, zbir);

    return 0;
}
```

```
Koliko prvih prirodnih brojeva treba sabrati?
6
i= 1 zbir= 1
i= 2 zbir= 3
i= 3 zbir= 6
i= 4 zbir= 10
i= 5 zbir= 15
i= 6 zbir= 21
Zbir prvih 6 prirodnih brojeva je 21
```