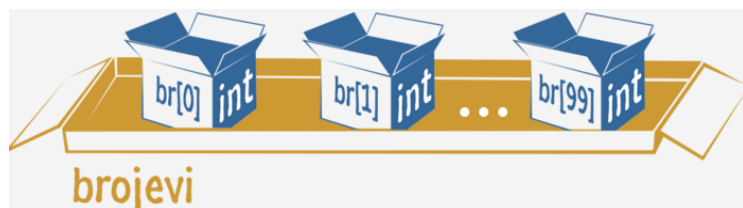


## Uvod u programiranje - nizovi



Uvežbavamo – Bubble Bee zadaci

<http://bee.bubblecup.org/Lekcije/nizovi>

Posetite site

<http://bee.bubblecup.org>

i registrujte se do sledećeg časa

### Domaći zadaci (priprema za pismeni)

1. Mladi čarobnjak Lehtos kasni na svoj prvi čarobnjački sastanak na ostrvu Asti.

Na sastanak, Lehtos mora da ponese drevni spis koji je negde zaturio i sada koristi poslednje minute da ga pronađe, pre nego što krene. Da bi stigao do ostrva, on najpre mora da se vozi kočijom do koje mu treba 15 minuta. Kočijom se vozi do luke 45 minuta, a potom se brodom vozi 1 sat i 12 minuta.

Kada stigne na ostrvo treba mu još 22 minuta da stigne na mesto sastanka.

Obzirom da na ostrvo Asti tog dana dolaze čarobnjaci iz čitavog sveta, kočije počinju da voze čarobnjake do luke od 5:00 ujutru i celog dana polaze na svakih **K** minuta. Takođe je angažovano nekoliko brodova, koji počev od 5:00 ujutru isplovljavaju iz luke ka ostrvu Asti na svakih **B** minuta.

Ako sastanak čarobnjaka počinje u **S** sati i **M** minuta, pri čemu je vreme početka između 9:00 i 23:59, napisati program koji za date vrednosti **K**, **B**, **S** i **M**, određuje kada tačno Lehtos mora da krene ka kočiji, da ne bi zakasnio na sastanak.

Primer: Ulaz: K = 35 B = 45 S = 14 M = 30 Izlaz: 11:10

2. Dat je prirodni broj **N**. Za prirodni broj **B** > 0 kažemo da je delilac broja **N** ukoliko **B** deli **N**. Napisati program SREDINA koji će ispisati aritmetičku sredinu svih delilaca datog broja **N**. Aritmetička sredina brojeva **B**<sub>1</sub>, **B**<sub>2</sub>, ..., **B**<sub>m</sub> jednaka je:  $(B_1 + B_2 + \dots + B_m) / m$ . Prvi i jedini red standardnog ulaza sadrži prirodni broj **N** ( $1 \leq N \leq 10^9$ ). U prvi i jedini red standardnog izlaza ispisati aritmetičku sredinu delilaca datog prirodnog broja. Broj štampati sa tačnošću od dve decimale.

PRIMER

ULAZ	IZLAZ
6	3.00

**Objašnjenje:** Delioци broja 6 su: 1, 2, 3 i 6. Njihova aritmetička sredina je:  $(1 + 2 + 3 + 6) / 4 = 3$ .

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

```
#include <math.h>
```

```
????? n,i,x,j,s;
```

```
float sred;
```

```
int main(){
```

```
scanf("%ld",&n);
```

```
s=0; //suma delilaca
```

```
i=0; // broj delilaca
```

```
x=floor(sqrtl(n)); // nalazenje korena (sqrtl-square root) broja n – sta je koren?
```

```
if(x*x==n){
```

```
s=s+x; //uvecanje sume delilaca, jer x deli n. Otkud to znamo?
```

```

i=i+1; //uvecanje broja delilaca za 1, jer x deli n
x=x-1; //smanjujemo kandidata za delioce
}

for(j=1;j<=x;++j) //delioce trazimo medju brojevima od 1 do sqrtl(n)
if(n%j==0){
i=i+2; //uvecanje broja delilaca za 2, jer j deli n, ali i n/j deli n
s=s+j+n/j; //uvecanje sume delilaca za j, n/j, jer j deli n, ali i n/j deli n
}

????? //aritmeticka sredina

//printf("%.2fn",sred); //preciznost ispisa u C-u
return 0;
}

```

C++ ispis decimalnih mesta  
cout << setprecision(2) << sred;

```

#include <iostream> // std::cout, std::fixed
#include <iomanip> // std::setprecision
using namespace std;

int main () {
double f=3.14159;
cout << setprecision(5) << f << '\n';
cout << setprecision(9) << f << '\n';
cout << std::fixed;
cout << setprecision(5) << f << '\n';
cout << setprecision(9) << f << '\n';
return 0;
}

```

Izlaz:

```

3.1416
3.14159
3.14159
3.141590000

```

3. Tri korisnika poslovne mreže prijavili su upotrebu interneta, za vreme neradne subote. Za svakog korisnika prijavljeno je vreme početka i završetka rada (sat i minut). Ako u nekom trenutku samo jedan korisnik koristi internet tada je zauzeta samo jedna pristupna linija, ukoliko su dva korisnika istovremeno priključena na internet, zauzete su dve linije, a ukoliko su sva tri korisnika istovremeno priključena, zauzete su sve tri linije. Radi proračuna dodatnih troškova firme vodi se evidencija o zauzetosti pristupnih linija. Napisati program koji za uneta vremena početka i završetka rada za sva tri korisnika (**PS1, PM1, ZS1, ZM1, PS2, PM2, ZS2, ZM2, PS3, PM3, ZS3, ZM3**) ispisuje koliko je vremena (sat i minut) bila zauzeta samo jedna linija, koliko su minuta bile zauzete dve linije, a koliko minuta su bile zauzete tri linije.

Primer:

Ulaz:

```

PS1 = 10 PM1 = 15 ZS1 = 16 ZM1 = 30
PS2 = 10 PM2 = 5 ZS2 = 15 ZM2 = 20
PS3 = 9 PM3 = 15 ZS3 = 12 ZM3 = 18

```

Izlaz: Jedna linija 2 h 0 min, dve linije 3 h 12 min, tri linije 2 h 3 min

## Nizovi

4. Niz 1, 1, 2, 3, 5, 8, 13, 21, 34, 55... se zove Fibonačijev niz.

Prva dva člana tog niza su jednaki 1, a svaki sledeći član se dobija kao zbir prethodna dva člana niza.

Napisati program koji sa ulaza učitava ceo broj n ( $n < 50$ ) i zatim formira niz prvih n Fibonačijevih brojeva.

Na primer: ako korisnik unese  $n=6$ , izlaz treba da bude niz 1, 1, 2, 3, 5, 8.

```
#include<stdio.h>
#define MAX_DUZINA 50

int main(){
    int f[MAX_DUZINA];
    int n;
    int i;

    /* učitavamo broj elemenata niza */
    printf("Unesite broj n: ");
    scanf("%d", &n);

    if(n<=0){
        printf("Niste uneli pozitivan broj!\n");
        return 0;
    }

    /* izracunavamo elemente niza */
    f[0]=1;
    f[1]=1;

    for(i=2;i<n;i++)
        f[i]=f[i-1]+f[i-2];

    /* ispisujemo elemente niza*/
    for(i=0; i<n; i++)
        printf("%d ", f[i]);
    printf("\n");

    return 0;
}
```

5. Napisati program kojim se učitava ceo broj n ( $n < 20$ ), dva celobrojna niza a i b duzine n i potom formira niz c naizmenicnim mesanjem elemenata nizova a i b.

Na primer:

ako korisnik unese  $n=5$  i

a: 1 2 5 6 9

b: 6 3 1 8 0

rezultujući niz je

c: 1 6 2 3 5 1 6 8 9 0

```

#include<stdio.h>
#define MAX_DUZINA 50

int main(){
    int a[MAX_DUZINA];
    int b[MAX_DUZINA];
    int c[2*MAX_DUZINA];
    int n;
    int i;

    /* učitavamo broj elemenata niza */
    printf("Unesite broj elemenata: ");
    scanf("%d", &n);

    if(n<=0){
        printf("Niste uneli pozitivan broj!\n");
        return 0;
    }

    /* učitavamo elemente niza a */
    printf("Unesite elemente niza a: ");
    for(i=0;i<n;i++)
        scanf("%d", &a[i]);

    /* učitavamo elemente niza b */
    printf("Unesite elemente niza b: ");
    for(i=0;i<n;i++)
        scanf("%d", &b[i]);

    /* formiramo niza c */
    for(i=0;i<n;i++){
        c[2*i]=a[i];
        c[2*i+1]=b[i];
    }

    /*
    II nacin

    for(i=0;i<2*n; i++)
        if(i%2==0)
            c[i]=a[i/2];
        else
            c[i]=b[i/2];

    */

    /* ispisujemo elemente niza c*/
    for(i=0; i<2*n; i++)
        printf("%d ", c[i]);
    printf("\n");

```

```
    return 0;
}
```

6. Napisati C program koji će učitati niz od 10 članova, a potom ispisati članove tog niza unazad tj. počevši od poslednjeg člana niza.

```
#include <stdio.h>
#define DIMENZIJA 10

int main() {
    int niz[DIMENZIJA], i;
    for (i = 0; i < DIMENZIJA; i++)
        scanf("%d", &niz[i]);
    printf("Ispisujemo brojeve unazad:\n");
    for (i = DIMENZIJA - 1; i >= 0; i--)
        printf("%d ", niz[i]);
    printf("\n");
    return 0;
}
```

7.

Napisati program koji učitava broj elemenata niza ( $n \leq 100$ ), zatim učitava elemente niza i ispisuje:

- elemente niza koji se nalaze na parnim indeksima
- parne elemente niza

TEST PRIMERI

Osmislimo sami!!!

```
#include <stdio.h>
#define MAX 100

int main()
{
    int a[MAX];
    int n;
    int i;

    printf("Unesi dimenziju niza:");
    scanf("%d", &n);

    for (i=0; i<n; i++)
    {
        printf("a[%d]=",i);
        scanf("%d", &a[i]); /* učitavamo jedan po jedan element niza */
    }

    printf("Elementi sa parnim indeksima:\n");
    for (i=0; i<n; i+=2)
        printf("a[%d]=%d\n",i,a[i]);

    printf("Parni elementi:\n");
```

```

for (i=0; i<n; i++)
    if (a[i]%2==0)
        printf("a[%d]=%d\n",i,a[i]);

return 0;
}

```

8. Napisati program koji učitava jedan ceo broj a zatim ispisuje koliko puta koja cifra ucestvuje u zapisu tog broja.

Nije potrebno ispisivati da se neka cifra pojavila 0 puta.

Na primer, za uneti broj 4611, izlaz treba da bude:

U zapisu broja 4611, cifra 1 se pojaviljuje 2 puta

U zapisu broja 4611, cifra 4 se pojaviljuje 1 puta

U zapisu broja 4611, cifra 6 se pojaviljuje 1 puta

A za uneti broj -252

U zapisu broja -252, cifra 2 se pojaviljuje 2 puta

U zapisu broja -252, cifra 5 se pojaviljuje 1 puta

```

#include<stdio.h>
#include<stdlib.h>
#define MAX 100

int main()
{
    int x;
    int brojac[10];
    char cifra;
    int original;
    int i;

    printf("Unesi jedan ceo broj:");
    scanf("%d",&x);

    /*
        svaki element niza brojac predstavlja
        brojac za jednu cifru:
        brojac[0] sadrzi broj nula
        brojac[1] sadrzi broj jedinica
        ...
        brojac[9] sadrzi broj devetki

        brojac se inicijalizuju na vrednost 0
    */

    for(i=0;i<10;i++)
        brojac[i]=0;

    /*
        vrednost promenljive x ce biti unistena
    */
}

```

```

u while petlji jer je u svakom koraku delimo
sa 10; njenu vrednost cuvamo u promenljivoj
original kako bismo mogli da je iskoristimo
na kraju prilikom ispisa
*/
original = x;

/*
Uzimamo apsolutnu vrednost broja za slucaj
da je uneti broj negativan
*/
x=abs(x);

/* Izdvajanje cifara broja */
do
{
cifra = x%10;
brojaci[cifra]++; /* Uvecavamo brojac odgovarajuce cifre */
x/=10;
} while(x);

/* Ispis brojaca koji su razliciti od nule */
for(i=0;i<10;i++)
if(brojaci[i])
printf("U zapisu broja %d, cifra %d se pojaviljuje %d puta\n", original, i, brojaci[i]);

return 0;
}

```

9.

. (Okružno 2015) Ako se članovi Fibonačijevog niza zapišu jedan pored drugog bez korišćenja razmaka i drugih simbola, dobija se sledeći Fibonačijev niz cifara 11235813213455... Deca Beograda se takmiče u pogađanju  $n$ -te cifre ovog niza cifara. Napisati program koji će ispisati  $n$ -tu cifru Fibonačijevog niza cifara.

**ULAZ:** U jedinom redu standardnog ulaza se nalazi jedan prirodan broj  $n$  ( $1 \leq n \leq 500$ )

**IZLAZ:** U jedinom redu standardnog izlaza ispisati  $n$ -tu cifru Fibonačijevog niza cifara.

**PRIMER**

ULAZ	IZLAZ
12	4
1	1
315	7
333	2
420	5
22	3
44	6
168	8

```

#include<iostream>
using namespace std;
int brojCifara(long long a)
{
int br=0;
while(a!=0)
{
a=a/10;
br++;
}
}

```

```

    }
    return br;
}

int main()
{
    long long f1=1, f2=1, f3; //clanovi Fibonacijevog niza
    int n,br=2,k,i;
    cin>>n;
    if(n==1||n==2) cout<<1<<endl;
    else
    {
        while (n>br)
        {
            n=n-br;
            //racunamo clanove niza
            f3=f1+f2; f1=f2; f2=f3;
            br=brojCifara(f3);
        }
        k=br-n;
        for(i=0;i<k;i++) f3=f3/10;
        cout<<f3%10<<endl;
    }
    return 0;
}

```

#### Domaći zadaci

10. Napisati program koji sa standardnog ulaza učitava dimenziju niza (broj manji od 100) i elemente niza, a zatim štampa niz u kojem su najveći i najmanji element niza razmenili mesta.

##### Test Primer 1

Unesite dimenziju niza: 5

Unesite elemente niza: 8 -2 11 19 4

Izlaz 8 19 11 -2 4

##### Test Primer 2

Unesite dimenziju niza: 10

Unesite elemente niza: 46 -2 51 8 -5 66 2 8 3 14

Izlaz 46 -2 51 8 66 -5 2 8 3 14

11. Sa standardnog ulaza se učitava dimenzija niza (broj manji od 100), elementi niza i jedan ceo broj k. Napisati program koji štampa indekse elemenata koji su deljivi sa k.

##### Test Primer 1

Unesite dimenziju niza: 4

Unesite elemente niza: 10 14 86 20

Unesite broj k: 5

Izlaz: 0 3

##### Test Primer 2

Unesite dimenziju niza: 4

Unesite elemente niza: 6 14 8 9

Unesite broj k: 5

Izlaz: U nizu nema elemenata koji su deljivi brojem 5!



### Test Primer 3

Unesite dimenziju niza: 6

Unesite elemente niza: 8 9 11 -4 8 11

Unesite broj k: 2

Izlaz: 0 3 4

12. Napisati program koji za dva cela broja x i y koja se učitavaju sa standardnog ulaza proverava da li se zapisuju pomoću istih cifara (jednak broj pojava jednakih cifara). Napomena: iskoristiti niz za čuvanje broja pojavljivanja svake od cifara.

### Test Primer 1

Unesite dva broja: 251 125

Brojevi se zapisuju istim ciframa!

### Test Primer 2

Unesite dva broja: 8898 9988

Brojevi se ne zapisuju istim ciframa!

### Primer 3

Unesite dva broja: -7391 1397

Brojevi se zapisuju istim ciframa!