

## Procesiranje teksta i upotreba ciklusa while, for, do while

```
EOF marker kraja ulaza iz <stdio.h>  
#define EOF (-1)
```

### DEMONSTRACIJA upotrebe funkcija iz zaglavlja stdio.h: getchar i putchar

```
int getchar (void);
```

**učitava znak sa standardnog ulaza, upotreba: znak=getchar();**

```
putchar (int znak)
```

**ispisuje znak na standardni izlaz**

#### *Primer 1.*

```
#include <stdio.h>  
main()  
{  
    int c1, c2;  
  
    c1 = getchar();  
    c2 = getchar();  
  
    printf("c1 = %d, c2 = %d\n",c1, c2);  
    printf("c1 = %c, c2 = %c\n",c1, c2);  
    putchar(c1); /* isto je kao i printf("%c",c1); */  
    putchar(c2); /* isto je kao i printf("%c",c2); */  
    putchar('\n');  
  
    /* Za ispisivanje karaktera a */  
    putchar('a');  
    /* dozvoljeno je : printf("abc"); printf("a"); */  
    /* nedozvoljeno je : printf('a'); putchar('abc'); putchar("abc"); */  
}
```

**Ulaz:**  
**ab**

**Izlaz:**  
**c1 = 97, c2 = 98**  
**c1 = a, c2 = b**  
**ab**  
**a**

**Primer 2.** Program cita jedan karakter i ispisuje ga.

```
#include <stdio.h>
main()
{
    int c; /* Karakter - obratiti paznju na int */
    c = getchar(); /* cita karakter sa standardnog ulaza */
    putchar(c); /* pise karakter c na standardni izlaz */
    putchar('\n'); /* pise prelazak u novi red */
    putchar('a'); /* ispisuje malo a */
    putchar(97); /* ekvivalentno prethodnom */
}
```

Ulaz:

s

Izlaz iz programa:

s

aa

48. Ispisati na standardni izlaz ukupan broj karaktera standardnog ulaza do markera kraja. Pretpostaviti da je za ukupan broj karaktera dovoljan opseg *double* promenljive.

```
#include <stdio.h>
main()
{
    double br;
    for( br=0; getchar()!=EOF; ++br );
    printf("Karaktera: %.0f\n",br);
}
```

49. NCP koji će ispisati na standardni izlaz ukupan brojbroj prelazaka u novi red u tekstu koji se učitava sa standardnog ulaza. Pretpostaviti da je za ukupan broj prelazaka u novi red dovoljan opseg *long* promenljivih.

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

```

int znak; /*prihvata znak sa ulaza */
long linije=0 ; /*brojac linija */
while ( (znak=getchar() ) != EOF)
if (znak=='\n') linije ++;

printf("Prelazaka u novi red: %ld\n",linije);
}

```

50. NCP koji će tekst ulazne datoteke prepisati u izlaznu datoteku tako da se višestruke uzastopne pojave znaka \* zamene jednom zvezdicom.

ulaz (stdin, getchar())	izlaz (stdout, putchar())
1*3[tab]** a***d efg*** [enter] ^Z	1*3[tab]* a*d efg*+* EOF

```

#include <stdio.h>
#define GRANICA '0'
main()
{
int znak; /*tekuci znak sa ulaza*/
int preth; /*znak koji prethodi tekucem */
preth=GRANICA;
while ( (znak=getchar() ) !=EOF)
{
if (znak !='*' || preth != '*') putchar(znak);
preth=znak;
}
}

```

#### 51. (naredba break)

Napisati C program koji proverava korektnu uparenost zagrada ( i ) u tekstu koji dolazi sa standardnog ulaza. Npr. zagrade su korektno uparene u izrazu 5 \* (8+3), a nisu u izrazu 3 + )2 ili izrazu ( (3+6) \* 8

```

#include <stdio.h>

main()
{
int c; /*tekuci karakter sa ulaza*/
int br_otv = 0; /*brojac zagrada*/

```

```

/*citanje teksta karakter po karakter do kraja ulaza (EOF) */
while((c=getchar()) != EOF)
{
    if(c=='(')    br_otv++;
    if (c== ')')
        { br_otv--;
          if (br_otv<0)
{ printf("Visak zatvorenih zagrada\n"); break; }
        }
    }

    if (br_otv == 0) printf("Zagrade su u redu\n");
    else if (br_otv >0) printf("Visak otvorenih
zagrada\n");
}

```

Naredba break se često koristi da se napusti beskonačna petlja. Pogledajmo petlju nalik na petlju iz prethodnog zadatka:

```

while(1)
{    c=getchar();
    if (c==EOF) brak;
    if(c=='(')    br_otv++;
    if (c== ')')
        { br_otv--;
          if (br_otv<0)
            { printf("Visak zatvorenih zagrada\n"); break; }
        }
    }
}

```

52. (naredba continue) NCP koji prihvata sa standardnog ulaza pozitivan ceo broj n ( $n \leq 50$ ), proverava korektnost unete vrednosti, a zatim učitava n celih brojeva. Ispisati sumu nenegativnih brojeva na standardni izlaz.

```

#include <stdio.h>
main()
{
    int n; /* dimenzija */
    int indeks; /* brojac */
    int suma, broj; /*suma nenegativnih clanova, ucitani broj*/
}

```

```

/*unos dimenzije n i proverava da li je taj broj u
segmentu 1..50 */
do
{ printf("Unesite broj elemenata niza\n");
scanf("%d", &n);
} while (n <1 || n > 50);
/* unos n celih brojeva i racunanje sume nenegativnih*/
for( indeks=0, suma=0; indeks<n; indeks++)
{
scanf("%d", &broj);
/* ignorisanje pozitivnih brojeva */
if(broj<0) continue;

suma+=broj; /*dodavanje nenegativnog broja sumi*/
}

printf("Suma nenegativnih je %d\n",suma);
}

```

Provezbavanje zadataka koji koriste cikluse

53. NCP koji ispisuje tabelu ascii kodova od 32-128.

```

#include <stdio.h>
main()
{
    int c;
for (c = 32; c<128; c++) printf("%d - %c\n",c,c);
}

```

54. NCP koji ucitava do 15 celih brojeva sa standardnog ulaza i pronalazi maksimum brojeva sa ulaza. Verzija bez niza

```

#include <stdio.h>
#define BR_ELEM 5
main()
{
    int a, max, i; /* broj sa ulaza, maksimum ucitanih brojeva, brojac u ciklusu*/
    scanf("%d",&a);
    max = a;
    for (i = 1; i < BR_ELEM; i++)
    {
        scanf("%d",&a);

```

```

if (a>max) max = a;
}
printf("Max : %d\n", max);
}

```

### 55. NCP koji racuna zbir

$$1 + x + \frac{x^2}{2} + \dots + \frac{x^n}{n!}$$

```

#include<stdio.h>
main()
{
    float f, suma; /* faktor sume, suma */
    float x; /* promenljiva x iz izraza */
    int i; /* brojac u petljama */
    int n; /* broj sabiraka */
    scanf("%d", &n);
    scanf("%d", &x);
    /* Pocetne inicijalizacije */
    f = 1; suma = 1;
    /* U jednom prolazu petlje dodajemo tekuci sabirak */
    for(i = 1; i <=n; i++) {
        f = f * x / i;
        suma = suma + f;
    }

    printf("Suma prvih %d clanova je %f \n", n, suma);
}

```

**56.NCP koji učitava sa standardnog ulaza osam prirodnih brojeva koji su rastuće uređeni. Program treba da proveri da li su učitani brojevi uzastopni celi brojevi tako da kad god postoje prekidi, ispisuje za svaki nedostajući opseg najmanji i najveći ceo broj koji nedostaje sa crticom između.**

ulaz	izlaz
5 6 7 11 12 13 15 18	8-10 14-14 16-17
2 3 4 5 6 7 8 9	

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

```

#define MAX 8
main()
{
    unsigned a,b,i;

    printf("Unesite brojeve: ");
    scanf("%u",&b);
    i=1;
    while(i<MAX)
    {
        a=b;
        scanf("%u",&b);
        i++;
        if(b!= (a+1)) printf("|%u-%u",a+1,b-1);
    }

    printf("|");
}

```

**57. NCP koji učitava prirodne brojeve  $n$ ,  $m$  i izračunava sumu kvadrata parnih brojeva od  $n$  do  $m$ .**

```

#include<stdio.h>
main()
{
    int n, m; /* donja i gornja granica intervala*/
    int i; /* paran broj iz intervala */
    long suma; /* zbir parnih*/
    scanf("%d%d", &n, &m);
    for(i=n, suma=0; i<=m; i++)
        if(i%2==0) suma+=i*i; //suma=suma+i*i;
    printf("\nZbir parnih je %ld\n", suma);
}

```