

Programiranje 1

Beleške sa vežbi

Smer *Informatika*

Matematički fakultet, Beograd

Jelena Tomašević

November 12, 2005

Sadržaj

1		5
1.1	Vežbe (7.11.2005.)	5

Čas 1

1.1 Vežbe (7.11.2005.)

Primer 1 Šta će biti izlaz iz sledećeg programa?

```
#include <stdio.h>

main()
{
    printf("\nZdravo, svima\n\n");
    printf("\n\tprelazak u novi red\n");
    printf("\t\ttabulator\n");
    printf("\\\\\\tkosa crta\n");
    printf("%%%\tprocenat\n");
}
```

```
Izlaz iz programa:
"Zdravo, svima"
\n prelazak u novi red
\t tabulator
\\ kosa crta
%% procenat
```

Primer 2 *A šta iz ovog?*

```
#include <stdio.h>

main()
{
    putchar('\\');
    putchar('t');
    putchar('\t');
    printf("Za %d ispisujem %c", '\\', '\\');
    printf("\n\n\\n\\n\\n\\n");
}
```

```
Izlaz iz programa:
\t Za 92 ispisujem \
```

```
\n\  
\n
```

Primer 3 *A iz ovog?*

```
#include <stdio.h>  
main()  
{  
    printf("Slova:\n%3c\n%5c\n", 'z' , 'Z');  
}
```

Izlaz iz programa:

```
Slova  
  z  
   Z
```

Primer 4

```
#include <stdio.h>  
main()  
{  
    int vrednost;  
    vrednost='A';  
    printf("%s\nkarakter=%3c\nvrednost=%3d\n",  
          "Veliko slovo",vrednost,vrednost);  
    vrednost='a';  
    printf("%s\nkarakter=%3c\nvrednost=%3d\n",  
          "Malo",vrednost,vrednost);  
}
```

Izlaz (u slucaju ASCII):

```
Veliko slovo  
karakter= A  
vrednost= 65  
Malo  
karakter= a  
vrednost= 97
```

Primer 5 *Program prikazuje unos i ispis realnih brojeva.*

```
#include <stdio.h>  
main()  
{  
    float x;  
    printf("Unesi realan broj : ");  
  
    /* Eksperimentisati sa vrednoscu npr. 34.56 */  
    scanf("%f",&x);  
    printf("Uneli ste broj (%f): %f\n", x);  
    printf("U naucnoj notaciji (%g): %g\n", x);  
}
```

Ulaz:
Unesi realan broj: 1.234
Izlaz iz programa:
Uneli ste broj (%f): 1.234000
U naucnoj notaciji (%g): 1.234

Primer 6 Program prikazuje zaokruživanje realnog broja prilikom ispisa.

```
#include <stdio.h>

main()
{
    float x;
    printf("Unesi realan broj : ");
    scanf("%f",&x);
    printf("Broj zaokruzen na dve decimale je : %.2f\n", x);
}
```

Ulaz:
Unesi realan broj: 1.234
Izlaz iz programa:
Broj zaokruzen na dve decimale je : 1.23

Primer 7 Program cita jedan karakter i ispisuje ga - demonstracija putchar i getchar.

```
#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');  /* prelazak u novi red */
    putchar('a');   /* ispisuje malo a */
    putchar(97);    /* ekvivalentno prethodnom */
}
```

Ulaz:
s
Izlaz iz programa:
s
s
aa

Primer 8 Program prepisuje standardni ulaz na standardni izlaz. Ilustracija redirekcije standardnog ulaza i izlaza : pokrenuti program sa :

```
./a.out <primer.c
./a.out >tekst.txt
./a.out <primer.c >kopija.c
```

```
/
#include <stdio.h>

main()
{
    int c;
    /* Obratiti paznju na raspored zagrada */
    while ((c = getchar()) != EOF)
        putchar(c);
}
```

Primer 9 *Demonstracija pretprocesorske direktive #define*

```
#include<stdio.h>
/* Racuna sumu dva broja */
#define sum(a,b) ((a)+(b))

/* Racuna kvadrat broja - pogresna verzija */
#define square_w(a) a*a

/* Racuna kvadrat broja */
#define square(a) ((a)*(a))

/* Racuna minimum tri broja */
#define min(a, b, c) (a)<(b) ? ((a)<(c) ? (a) : (c)) : ((b)<(c) ? (b) : (c))

main()
{
    printf("sum(3,5) = %d\n", sum(3,5));
    printf("square_w(5) = %d\n", square_w(5));
    printf("square_w(3+2) = %d\n", square_w(3+2));
    printf("square(3+2) = %d\n", square(3+2));
    printf("min(1,2,3) = %d\n", min(1,2,3));
    printf("min(1,3,2) = %d\n", min(1,3,2));
    printf("min(2,1,3) = %d\n", min(2,1,3));
    printf("min(2,3,1) = %d\n", min(2,3,1));
    printf("min(3,1,2) = %d\n", min(3,1,2));
    printf("min(3,2,1) = %d\n", min(3,2,1));
}
```

Izlaz iz programa:

```
sum(3,5) = 8
square_w(5) = 25
square_w(3+2) = 11
square(3+2) = 25
min(1,2,3) = 1
min(1,3,2) = 1
min(2,1,3) = 1
```



```
min(2,3,1) = 1
min(3,1,2) = 1
min(3,2,1) = 1
```

Primer 10 *Demonstracija sizeof operatora. sizeof operator izračunava veličinu tipa odnosno promenjive.*

```
#include<stdio.h>
main()
{
    int i;
    float f;
    int n[10];

    printf("sizeof(int)=%d\n", sizeof(int));
    printf("sizeof(long)=%d\n", sizeof(long));
    printf("sizeof(short)=%d\n", sizeof(short));
    printf("sizeof(signed)=%d\n", sizeof(signed));
    printf("sizeof(unsigned)=%d\n", sizeof(unsigned));
    printf("sizeof(char)=%d\n", sizeof(char));
    printf("sizeof(float)=%d\n", sizeof(float));
    printf("sizeof(double)=%d\n", sizeof(double));

    printf("sizeof(i)=%d\n", sizeof(i));
    printf("sizeof(f)=%d\n", sizeof(f));
    printf("sizeof(n)=%d\n", sizeof(n));
    printf("Broj elemenata niza n : %d\n", sizeof(n)/sizeof(int));
}
```

Izlaz iz programa(u konkretnom slucaju):

```
sizeof(int)=4
sizeof(long)=4
sizeof(short)=2
sizeof(signed)=4
sizeof(unsigned)=4
sizeof(char)=1
sizeof(float)=4
sizeof(double)=8
sizeof(i)=4
sizeof(f)=4
sizeof(n)=40
Broj elemenata niza n : 10
```

Primer 11 *Program vrši prebrojavanje cifara unetih na ulazu.*

```
#include <stdio.h>

/* zbog isdigit */
```

```
#include <ctype.h>
main()
{
    int c;
    int br_cifara = 0;
    while ((c = getchar()) != EOF)
        if ('0'<=c && c<='9') /* moze i if (isdigit(c)) */
            br_cifara++;

    printf("Broj cifara je : %d\n", br_cifara);
}
```

Primer 12 Program vrši brojanje pojavljivanja karaktera 0, 1 i 2 (ilustruje switch).

```
#include <stdio.h>

main()
{
    int c;
    int br_0=0, br_1=0, br_2=0;

    while ((c = getchar()) != EOF)
    {
        switch(c)
        {
            /* Obratiti paznju da nije case 0: niti case '0'; */
            case '0':
                br_0++;
                break; /* Isprobati veziju bez break */
            case '1':
                br_1++;
                break;
            case '2':
                br_2++;
                break;
        }
    }
    printf("Br 0 : %d\nBr 1 : %d\nBr 2 : %d\n",br_0, br_1, br_2);
}
```

Primer 13 Program broji linije i znakove na ulazu.

```
#include <stdio.h>
main()
{
    int znak; /*prihvata znak sa ulaza */
    long linije=0 ; /*brojac linija */
    long br_znak=0; /*brojac znakova na ulazu */
}
```

```

while ( (znak=getchar() ) != EOF)
{
    br_znak++;
    if (znak=='\n') linije ++;
}

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

```

Primer 14 Program broji blankove, horizontalne tabulatore i linije na ulazu.

```

#include <stdio.h>
main()
{
    int znak; /*prihvata znak sa ulaza */
    int Blanks=0; /*brojac blankova */
    int Tabs=0; /*brojac horizontalnih tabulatora */
    int NewLines=0; /*brojac linija */

    /*ucitavanje karakter do markera kraja fajla i ...*/
    while( (znak=getchar())!=EOF )
        /* ... brojanje blankova */
        if( znak==' ' ) ++Blanks;
        /* ... brojanje tab-ova */
        else if( znak=='\t' ) ++Tabs;
        else if( znak=='\n' ) ++NewLines;

    /*UOCITI: blok naredbi while ciklus NIJE OGRADJEN viticastim zagradama*/

    /*izdavanje rezultata na standardni izlaz*/
    printf("Blankova: %d. Tabulatora: %d. Prelazaka u novi red: %d\n", Blanks, Tabs, NewLines);
}

```

Primer 15 Prepisuje ulaz na izlaz čineći tabulatore, nove linije i backslash-ove vidljivim.

```

#include <stdio.h>
main()
{
    int znak;
    znak=getchar();
    while( znak!=EOF )
    {
        if( znak=='\t' ) /*uciniti tab vidljivim */
            { putchar('\\'); putchar('t'); }
        else if( znak=='\n' ) /*uciniti new line vidljiv */

```

```

        { putchar('\\'); putchar('n'); putchar('\n'); }
        else if( znak=='\\' ) /*backslash udvojiti */
        { putchar('\\'); putchar('\\'); }
        else putchar(znak);

        znak=getchar();
    } /* while( znak!=EOF ) */

} /*main() */

```

Primer 16 Program koji prepisuje ulaz na izlaz pri čemu više blanko znakova zamenjuje jednim.

```

#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;
    }
}

```

Primer 17 Program proverava da li su zagrade (i) dobro uparene.

```

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

main()
{
    int c;
    int br_otv = 0;
    while((c=getchar()) != EOF)
    {
        switch(c)
        {
            case '(':
                br_otv++;
                break;
            case ')':
                br_otv--;
                if (br_otv<0)

```

```
        {
            printf("Visak zatvorenih zagrada\n");
            exit(1);
        }
    }

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

Primer 18 Program ilustruje korišćenje statičkih nizova. Ispisuje 10 unetih brojeva unazad.

```
#include <stdio.h>

main()
{
    int a[10];
    int i;
    for (i = 0; i<10; i++)
    {   printf("a[%d]=",i);
        scanf("%d",&a[i]);
    }

    printf("Unazad : \n");

    for (i = 9; i>=0; i--)
        printf("a[%d]=%d\n",i,a[i]);
}
```

Primer 19 Program pronalazi maksimum brojeva sa ulaza - verzija sa nizom.

```
#include <stdio.h>
#define BR_ELEM 5
main()
{
    int a[BR_ELEM];
    int i;
    int max;

    /* Ucitavamo niz brojeva */
    for (i = 0; i < BR_ELEM; i++)
        scanf("%d",&a[i]);

    /* Pronalazimo maksimum */
    max = a[0];
    for (i = 1; i < BR_ELEM; i++)
        if (a[i]>max)
            max = a[i];

    /* Ispisujemo maksimum */
    printf("Max = %d\n",max);
}
```

Primer 20 Program pronalazi maksimum brojeva sa ulaza - verzija bez niza.

```
#include <stdio.h>
#define BR_ELEM 5
```

```
main()
{
    int a, max, i;
    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);
}
```

Primer 21 *Brojanje pojavljivanja svake od cifara. Koriscenje niza brojača.*

```
#include <stdio.h>
#include <ctype.h>
main()
{
    /* Niz brojaca za svaku od cifara */
    int br_cifara[10];
    int i, c;

    /* Resetovanje brojaca */
    for (i = 0; i < 10; i++)
        br_cifara[i] = 0;

    /* Citamo sa ulaza i povecavamo odgovarajuce brojace */
    while ((c = getchar()) != EOF)
        if (isdigit(c))
            br_cifara[c-'0']++;

    /* Ispis rezultata */
    for (i = 0; i < 10; i++)
        printf("Cifra %d se pojavila %d put%s\n",
            i, br_cifara[i], br_cifara[i]==1?"":"a");
}
```

Primer 22 *Program ilustruje inicijalizaciju nizova.*

```
#include <stdio.h>

main()
{
```

```

/* Niz inicijalizujemo tako sto mu navodimo vrednosti
   u viticasnim zagradama. Dimenzija niza se odredjuje
   na osnovu broja inicijalizatora */
int a[] = {1, 2, 3, 4, 5, 6};

/* Isto vazi i za niske karaktera */
char s[] = {'a', 'b', 'c'};

/* Ekvivalentno prethodnom bi bilo
char s[] = {97, 98, 99};
*/

/* Broj elemenata niza */
int a_br_elem = sizeof(a)/sizeof(int);
int s_br_elem = sizeof(s)/sizeof(char);

/* Ispisujemo nizove */

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

for (i = 0; i < s_br_elem; i++)
    printf("s[%d]=%c\n",i, s[i]);

}

```

Primer 23 Program uvodi niske karaktera terminisane nulom.

```

#include <stdio.h>

main()
{
    /* Poslednji bajt niske karaktera se postavlja na '\0' tj. 0 */
    char s[] = {'a', 'b', 'c', '\0' };

    /* Kraci nacin da se postigne prethodno */
    char t[] = "abc";

    /* Ispis niske s karakter po karakter*/
    int i;
    for (i = 0; s[i] != '\0'; i++)
        putchar(s[i]);
    putchar('\n');

    /* Ispis niske s koristeći funkciju printf */
    printf("%s\n", s);
}

```



```
/* Ispis niske t karakter po karakter*/  
for (i = 0; t[i] != '\0'; i++)  
    putchar(t[i]);  
putchar('\n');  
  
/* Ispis niske t koristeci funkciju printf */  
printf("%s\n", t);  
  
}
```

Primer 24 `to_upper`, `to_lower` - funkcije konverzije u velika, odnosno mala slova.

```
#include <stdio.h>

/* Funkcija konvertuje mala slova u velika,
   a ostala ne menja */
char to_upper(char c)
{
    if ('a'<=c && c<='z')
        return c-'a'+'A';
    return c;
}

/* Funkcija konvertuje velika slova u mala,
   a ostala ne menja */
char to_lower(char c)
{
    if ('A'<=c && c<='Z')
        return c-'A'+'a';
    return c;
}

main()
{
    int c;
    while ((c = getchar()) != EOF)
        putchar(to_upper(c));
}
```