

## 1. međuispit iz Programiranja i programskog inženjerstva - 08.11.2005.

1. Napisati C-program u kojem treba s tipkovnice pročitati duljinu nekog konopca (cijeli broj). Konopac zatim treba razrezati na tri dijela, s tim da se njihove duljine odnose kao 2 : 3 : 4. Ispisati pročitane duljinu konopca, a u novom retku duljine dobivenih odsječaka. Napomena: duljine pojedinog dijela konopca mogu biti realni brojevi. (250 bodova)

```
#include <stdio.h>

int main(){
    int duljina;
    float a,b,c;
    printf("Unesite duljinu konopca:");
    scanf("%d",&duljina);
    a = 2 * duljina / 9.f ;
    b = 3 * duljina / 9.f ;
    c = 4 * duljina / 9.f ;
    printf("Duljine su: %f %f i %f\n",a,b,c);

    return 0;
}
```

2. Napisati C-program koji treba izračunati rješenje sustava jednačbi  $y=ax+b$  i  $y=ax^2+x$ . Vrijednosti brojeva  $a$  i  $b$  (realni brojevi) učitati s tipkovnice. Ukoliko je vrijednost broja  $a$  jednaka 0 ispisati poruku "a nije valjan" i završiti izvođenje programa. Rješenje mora biti kompletno, tj. mora obuhvatiti i slučaj kada su rješenja sustava konjugirano kompleksni brojevi. (250 bodova)

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

int main(){
    float a,b,x1,y1,x2,y2,xim1,xim2,yim1,yim2;
    float D;
    printf("Unesite a i b u jednačbama y=ax+b i y=ax^2+x:");
    scanf("%f %f",&a,&b);

    if (a==0){
        printf("a nije valjan");
    }
    else{

        /* računam diskriminantu */
        D=(1-a)*(1-a) + 4*a*b;
        if (D>0){
            x1 = ( (a-1) + sqrt(D) ) / (2 * a);
            x2 = ( (a-1) - sqrt(D) ) / (2 * a);
            y1 = a * x1 + b;
            y2 = a * x2 + b;
            printf("x1=%f, y1=%f\n",x1,y1);
            printf("x2=%f, y2=%f\n",x2,y2);
        }
        else if (D==0){
            x1 = ( a-1 ) / (2 * a);
            y1 = a * x1 + b;
            printf("x=%f, y=%f\n",x1,y1);
        }
        else{
            x1 = ( (a-1) ) / (2 * a); /* realni dio prvog broja */

```

```

        xim1 = sqrt(-D) / (2 * a); /*imaginarni dio prvog broja*/
        y1 = a * x1 + b;
        yim1 = a * xim1;

        x2 = ( (a-1) ) / (2 * a); /*realni dio drugog broja */
        xim2 = -sqrt(-D) / (2 * a); /*imaginarni dio drugog broja*/
        y2 = a * x2 + b;
        yim2 = a * xim2;

        printf("x1=%f+i*f, y1=%f+i*f\n",x1,xim1,y1,yim1);
        printf("x2=%f+i*f, y2=%f+i*f\n",x2,xim2,y2,yim2);
    }
}
return 0;
}

```

3. Napisati C-program koji će s tipkovnice pročitati neki cijeli četveroznamenasti broj x. Ukoliko broj nije četveroznamenast, program treba završiti porukom "Broj nije četveroznamenast!". Ako je broj ispravno zadan, treba učitati cijeli broj n za kojeg mora vrijediti  $1 \leq n \leq 3$ . Ako n nije dobro zadan završiti program uz poruku "n nije valjan". Ukoliko su x i n valjani treba od x načiniti novi broj tako da mu se znamenke ciklički pomaknu za n mjesta u lijevo (na primjer, za n=2 i zadani broj x=1234 treba nakon posmaka dobiti broj 3412, za n=3 i x=1234 treba dobiti 4123). Ispisati pročitani broj, a u novom retku novonastali broj. Ukoliko je posmakom znamenaka to prestao biti četveroznamenasti broj, pokraj njega treba još dodati napomenu " - broj nije četveroznamenast". (250 bodova)

```

#include <stdio.h>
#include <math.h>

int main(){
    int x,n,novi;
    printf("Unesite četveroznamenasti broj: ");
    scanf("%d", &x);
    if (x<1000 || x>9999){
        printf("Broj nije četveroznamenast\n");
    }
    else{
        printf("Unesite n: ");
        scanf("%d", &n);
        if (n<1 || n>3){
            printf("n nije valjan");
        }
        else{
            novi = (x % (int) pow(10,4-n)) * pow(10,n) +
                    (x / (int) pow(10,4-n));
            printf("Stari broj=%d\n",x);
            printf("Novi broj=%d",novi);
            if (novi<1000)
                printf("-broj nije četveroznamenast\n");
        }
    }
    return 0;
}

```

4. Napisati C-program u kojem treba s tipkovnice pročitati 3 cijela broja koji predstavljaju dan, mjesec i godinu nekog datuma. Zatim s tipkovnice treba pročitati redni broj dana u tjednu kojim započinje dani mjesec (koristiti oznaku 1 za ponedjeljak, 2 za utorak itd.). Učitani datum treba ispisati tako da se najprije slovima ispiše dan u tjednu, a nakon toga treba ispisati datum tako da se iza brojeva kod ispisa doda točka. Na primjer, za unesene brojeve 9 11 2005 i 2 kao redni broj dana u tjednu kojim započinje dani mjesec, program treba ispisati: srijeda, 9.11.2005.g.  
Napomena: Možete pretpostaviti da su unesene vrijednosti ispravne. (250 bodova)

*Varijanta 1:*

```
#include <stdio.h>

int main(){
    int d,m,g,rbr,dpp,temp;
    printf("Unesite dan mjesec i godinu: ");
    scanf("%d %d %d", &d, &m, &g);
    printf("Unesite redni broj dana kojim pocinje mjesec: ");
    scanf("%d", &rbr);
    /*datum prvog ponedjeljka u mjesecu*/
    dpp = (9-rbr) % 7;
    /*udaljenost od ponedjeljka*/
    /* +7 je da se izbjegne slučaj kad je d<dpp */
    temp = (d+7-dpp) % 7;

    if (temp==0){
        printf("Ponedjeljak,");
    }
    else if (temp==1){
        printf("Utorak,");
    }
    else if (temp==2){
        printf("Srijeda,");
    }
    else if (temp==3){
        printf("Cetvrtak,");
    }
    else if (temp==4){
        printf("Petak,");
    }
    else if (temp==5){
        printf("Subota,");
    }
    else {
        printf("Nedjelja,");
    }

    printf("%d.%d.%d.g\n",d,m,g);

    return 0;
}
```

*Varijanta 2:*

```
#include <stdio.h>

int main(){
    int d,m,g,rbr,dan;
    printf("Unesite dan mjesec i godinu: ");
    scanf("%d %d %d", &d, &m, &g);
    printf("Unesite redni broj dana kojim pocinje mjesec: ");
```

```

scanf("%d", &rbr);
/*zamislim da mjesec ne počinje s danim danom već s ponedjeljkom
   pa su, zbog toga, svi datumi
   pomaknuti za udaljenost od ponedjeljka (rbr-1)*/
dan = (d+rbr-1) % 7;

if (dan==1){
    printf("Ponedjeljak,");
}
else if (dan==2){
    printf("Utorak,");
}
else if (dan==3){
    printf("Srijeda,");
}
else if (dan==4){
    printf("Cetvrtak,");
}
else if (dan==5){
    printf("Petak,");
}
else if (dan==6){
    printf("Subota,");
}
else {
    printf("Nedjelja,");
}

printf("%d.%d.%d.g\n",d,m,g);

return 0;

}

```

5. Napisati program u kojem treba pročitati tri znakovne varijable koje predstavljaju znamenke nekog broja. Za svaku od učitanih vrijednosti treba provjeriti je li uistinu znamenka i ako sve zadovoljavaju dani uvjet treba iz njih izračunati i ispisati vrijednost troznamenkastog broja kojeg reprezentiraju, pri čemu znamenka koja je prva učitana ima najveću težinu. (250 bodova)

```

#include <stdio.h>

int main(){
    char a,b,c;int broj;
    printf("Unesite 3 znamenke(bez razmaka) i pritisnite enter:");
    scanf("%c%c%c",&a,&b,&c);
    if (a>='0' && a<='9' && b>='0' && b<='9' && c>='0' && c<='9'){
        broj = 100 * (a-'0') + 10 * (b-'0') + c - '0';
        printf("Broj = %d\n", broj);
    }
    return 0;

}

```

6. Što će i na koji način ispisati slijedeći programski odsječak (jasno označiti lijevi rub ispisa na zaslonu, a znakom \_ označite eventualne praznine): (250 bodova)

```
float b, e, r=2.f;
short int x=1, d=0;
char a='\0', c;
a = 'a' - 32;
b = (float)(4/3*3)+4/3*3.f;
c = a + 32 >= 'A';
d = 32767+2;
e = 1/(int)r;
if(a=='\0')
    printf("%c\n", a);
else
    printf("%c %d\n", a, a);
printf("%f %d\n", b, c);
printf("%d %f\n", d, e);
```

Rješenje: ispis na ekranu je:

```
A 65
6.000000 1
-32767 0.000000
```

Kratko objašnjenje:

Nakon `a = 'a' - 32`, `a` postane 'A' ('a' -32 je isto što i 97 - 32 tj. 65 što je slovo 'A').

Kod `(float)(4/3*3)+4/3*3.f`, `4/3` je cjelobrojno djeljenje pa nije 1.3333 nego 1.

`c = a + 32 >= 'A'`; `a` je u tom trenutku jednak 'A' tj. 65, pa je izraz jednak `97 >= 65` što je istina, pa `c` postaje 1.

Prilikom `d = 32767+2` imamo 0111 1111 1111 1111 + 0000 0000 0000 0010 što je jednako 1000 0000 0000 0001. Obzirom da je `d` short broj, dakle sadrži i pozitivne i negativne brojeve, koristi se tehnika dvojnog komplementa pa je `d` jednak -32767.

`1/(int) r` Vrijednost broja `r` se unutar izraza privremeno pretvara u cijeli broj te dolazi do cjelobrojnog djeljenja 1/2 što je jednako 0.

Napomena: Potpuno je svejedno da li je u rješenju napisano 6.000000, 6, 6.0 ili neka slična varijanta.