

JMBAG	
Ime i prezime	

Programiranje i programsko inženjerstvo Međuispit

26. studenoga 2015.

**Rješenja 1, 2, 3 i 4. zadatka napisati na vlastitim papirima i predati ih u košuljici.
Rješenje 5. i 6. zadatka napisati na za to predviđeno mjesto uz zadatak.**

1. (5 bodova)

S tipkovnice učitati niz znakova koji sigurno sadrži samo velika slova engleske abecede i sigurno neće biti dulji od 100 znakova, te dva ne-negativna cijela broja a i b . Nije potrebno provjeravati ispravnost učitanih podataka. Pomoću tzv. Afinog kriptiranja načiniti **novi**, kriptirani niz. U novom, kriptiranom nizu, svaki znak ulaznog niza bit će predstavljen znakom dobivenim pomoću sljedeće funkcije:

$f(x) = (a * x + b) \% 26$, pri čemu je x redni broj slova u engleskoj abecedi, a počinje vrijednošću 0 ($x=0$ za 'A', $x=1$ za 'B', itd.).

Ispisati prvo novi, kriptirani niz znakova, a zatim ulazni niz znakova. Ispis mora biti usklađen s oblikom ispisa prikazanim u primjeru.

Primjer: Upisite niz: ABACUS
Upisite dva broja: 3 5
Kriptirani niz: FIFLNH
Ulazni niz: ABACUS

Komentar uz primjer: uz $a = 3$ i $b = 5$, slovo B (redni broj 1 u engleskoj abecedi) iz ulaznog niza kriptirano je kao slovo I jer je $(3 * 1 + 5) \% 26 = 8$, a slovo s rednim brojem 8 u engleskoj abecedi je I.

2. (6 bodova)

Brojevi bankovnih računa (cijeli brojevi) na koje se uplaćuju iznosi donacija (realni brojevi) su iz intervala [10000, 11000]. Na početku su računi prazni tj. iznos na svim računima je 0 kuna.

Učitavati parove brojeva: broj računa i iznos uplate. Nije potrebno provjeravati ispravnost učitanih podataka. Učitavanje prekinuti onda kada broj uplata na bilo kojem od računa dosegne 3 uplate. Nakon toga ispisati najveću sumu uplata po bilo kojem računu i sve brojeve računa s takvom (najvećom) sumom uplata. Ispis mora biti usklađen s oblikom ispisa prikazanim u primjeru.

Upisujte uplate (br.racuna i iznos):
10015 10.0
10006 170.10
10001 150.40
10002 230.50
10550 250.55
10015 77.90
10002 20.05
10015 30.05
Najveća suma uplata je 250.55
Brojevi racuna s tim iznosom su:
10002
10550

3. (5 bodova)

Napisati program koji će svaki član dvodimenzijskog polja znakova od 8 redaka i 8 stupaca postaviti na slovo K, zatim učitati cijele brojeve a i b koji će sigurno biti iz intervala [0, 7]. Nije potrebno provjeravati ispravnost učitanih podataka. Član koji se nalazi u retku a i stupcu b treba postaviti na slovo T, a sve ostale članove u retku a i stupcu b postaviti na znak minus (-). Polje ispisati na zaslon, pri čemu ispis mora biti usklađen s oblikom ispisa prikazanim u primjeru.

Upisite a i b: 2 5
K K K K K - K K
K K K K K - K K
- - - - - T - -
K K K K K - K K
K K K K K - K K
K K K K K - K K
K K K K K - K K
K K K K K - K K

4. (5 bodova)

Učitati pozitivne realne brojeve a , b i cijeli broj n . Nije potrebno provjeravati ispravnost učitanih podataka. Brojevi a i b predstavljaju početne članove nizova A i B (a_1 odnosno b_1). Član niza a_i izračunava se kao aritmetička sredina prethodnog člana niza A i prethodnog člana niza B , tj.

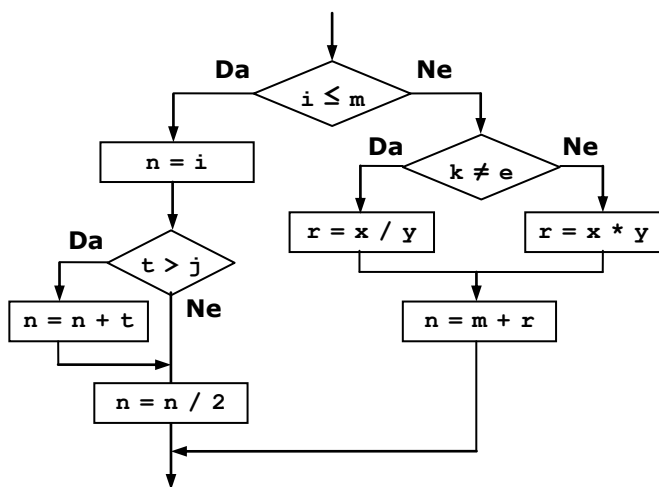
$a_i = \frac{a_{i-1} + b_{i-1}}{2}$. Član niza b_i izračunava se kao geometrijska

sredina prethodnog člana niza A i prethodnog člana niza B , tj. $b_i = \sqrt{a_{i-1} * b_{i-1}}$. Članove nizova A i B , od a_2 do a_n , te b_2 do b_n ispisati u skladu s oblikom ispisa prikazanim u primjeru.

Upisite a , b i n : 1. 5000. 5	
$A(2)=2500.5000$	$B(2)= 70.7107$
$A(3)=1285.6053$	$B(3)= 420.4903$
$A(4)= 853.0478$	$B(4)= 735.2446$
$A(5)= 794.1462$	$B(5)= 791.9588$

5. (2 boda)

Napisati **odsječak** C programa kojim će se realizirati dijagram toka prikazan na slici.



Prostor za odgovor

6. (2 boda)

Sadržaj registra u kojem je prema standardu IEEE 754 u jednostrukoj preciznosti pohranjen realni broj -6.6875 napisati u **oktalnom** obliku.

Prostor za odgovor

Rješenja

- ```
#include <stdio.h>
#define MAXZNAKOVA 100
int main(void) {
 char ulaz[MAXZNAKOVA + 1], izlaz[MAXZNAKOVA + 1];
 int a, b, i = 0;
 printf("Upisite niz: ");
 gets(ulaz);
 printf("Upisite dva broja: ");
 scanf ("%d %d", &a, &b);

 while (ulaz[i] != '\0') {
 izlaz[i] = ((ulaz[i] - 'A') * a + b) % 26 + 'A';
 ++i;
 }
 izlaz[i] = '\0';
 printf("Kriptirani niz: %s\n", izlaz);
 printf("Ulazni niz: %s\n", ulaz);
 return 0;
}
```
- ```
#include <stdio.h>
#define D_GR 10000
#define G_GR 11000
#define MAX_UPLATA_PO_RACUNU 3

int main (void) {
    int brojUplataNaRacun[G_GR - D_GR + 1] = {0};
    float sumaUplataNaRacunu[G_GR - D_GR + 1] = {0.f};
    int brRac;
    float iznos, najvecaSuma;
    printf("Upisujte uplate (br.racuna i iznos):\n");
    /* učitavati dok broj uplata na nekom racunu ne dosegne maksimum */
    do {
        scanf("%d %f", &brRac, &iznos);
        ++brojUplataNaRacun[brRac - D_GR];
        sumaUplataNaRacunu[brRac - D_GR] += iznos;
    } while (brojUplataNaRacun[brRac - D_GR] < MAX_UPLATA_PO_RACUNU);
    /* odrediti najvecu sumu na racunima */
    najvecaSuma = sumaUplataNaRacunu[0];
    for (brRac = D_GR+1; brRac <= G_GR; ++brRac) {
        if (najvecaSuma < sumaUplataNaRacunu[brRac - D_GR]) {
            najvecaSuma = sumaUplataNaRacunu[brRac - D_GR];
        }
    }
    /* ispisati sve racune kojima je suma jednaka najvecoj sumi */
    printf("Najveca suma uplata je %8.2f\n", najvecaSuma);
    printf("Brojevi racuna s tim iznosom su:\n");
    for (brRac = D_GR; brRac <= G_GR; ++brRac) {
        if (sumaUplataNaRacunu[brRac - D_GR] == najvecaSuma) {
            printf("    %5d\n", brRac);
        }
    }
    return 0;
}
```

```

3.  #include <stdio.h>
    #define N 8
    int main(void) {
        char m[N][N];
        int i, j, a, b;

        for (i = 0; i < N; ++i)
            for (j = 0; j < N; ++j)
                m[i][j] = 'K';

        printf("Unesite a i b: ");
        scanf("%d %d", &a, &b);
        for (i = 0; i < N; ++i)
            m[i][b] = m[a][i] = '-';

        m[a][b] = 'T';

        for (i = 0; i < N; ++i) {
            for (j = 0; j < N; ++j)
                printf("%c ", m[i][j]);
            printf("\n");
        }
        return 0;
    }

```

```

4.  #include <stdio.h>
    #include <math.h>
    int main(void) {
        float a, b, novi_a, novi_b;
        int i, n;
        printf("Upisite a, b i n: ");
        scanf("%f %f %d", &a, &b, &n);
        for (i = 2; i <= n; ++i) {
            novi_a = (a + b) / 2;
            novi_b = sqrt(a * b);
            printf("A(%d)=%9.4f B(%d)=%9.4f\n", i, novi_a, i, novi_b);
            a = novi_a;
            b = novi_b;
        }
        return 0;
    }

```

```

5.  if (i <= m) {
        n = i;
        if (t > j) {
            n = n + t;
        }
        n = n / 2;
    }
    else {
        if (k != e) {
            r = x / y;
        }
        else {
            r = x * y;
        }
        n = m + r;
    }

```

```

6.  300 654 000 00

```