

JMBAG	
Ime i prezime	

Programiranje i programsko inženjerstvo

Jesenski ispitni rok 13. rujna 2016.

Rješenja zadataka 1., 2. a) i 3. potrebno je napisati na vlastitim papirima i predati u košuljici.

Rješenja zadataka 2. b), 4., 5. i 6. napisati na za to predviđeno mjesto uz zadatak i predati u košuljici.

Zadatak 1. (20 bodova)

U postojećoj tekstualnoj datoteci `ocjene.txt` zapisani su podaci o ocjenama koje su studenti ostvarili na ispitima. Svaki zapis sadrži datum ocjenjivanja (do 10 znakova), JMBAG studenta koji je pristupio ispitu (uvijek 10 znakova), šifru predmeta (broj iz [1,9999]) i ocjenu koju je student ostvario na ispitu (1 broj).

Primjer zapisa da je student s JMBAG-om 003649123456 danas dobio izvrstan na predmetu sa šifrom 200:

```
2016-09-13 0036491234 200 1
```

U direktnoj binarnoj datoteci `predmeti.bin` pohranjeni su osnovni podaci o predmetima – svaki zapis sadrži šifru (`int`) i naziv (`char[95+1]`) predmeta. Redni broj zapisa u datoteci odgovara šifri predmeta. Potrebno je napisati funkciju

```
void statistika(char *predmet, char *datum);
```

koja će za predmet čije je ime zadano argumentom `predmet` ispisati koliko je studenata na zadani datum dobilo koju ocjenu. Argument `datum` je istog formata kao i datumi u tekstualnoj datoteci. Primjer očekivanog ispisa za poziv `statistika("Programiranje i programsko inženjerstvo", "2016-09-13");`

```
Broj ocjenjenih: 65
Izvrstan: 40
Vrlo dobar: 20
Dobar: 4
Dovoljan: 1
Nedovoljan: 0
```

Zadatak 2. (20 bodova)

a) Potrebno je napisati funkciju `lok`, koja prima cjelobrojno dvodimenzijско polje i koordinate jednog od elemenata tog polja. Funkcija vraća cijeli broj 1 ako je lokalni prosjek elementa na zadanim koordinatama manji od prosjeka svih elemenata polja, inače vraća 0. Lokalni prosjek elementa polja je aritmetička sredina vrijednosti svih susjeda tog elementa (horizontalno, vertikalno i dijagonalno) uključujući i vrijednost zadanog elementa

Primjer:

4	6	1	8	1
1	5	4	3	5
3	2	8	9	4
1	7	7	2	6

Za zadani element na koordinatama (0,0), lokalni prosjek računa se za vrijednost elementa i njegova 3 susjedna elementa. Lokalni i ukupan prosjek jednak je vrijednostima 4 i 4.35 stoga funkcija vraća vrijednost 1.

Za zadani element na koordinatama (2, 3), lokalni prosjek računa se za vrijednost elementa i njegovih 8 susjednih elemenata. Lokalni i ukupan prosjek jednak je vrijednostima 6 i 4.35 stoga funkcija vraća vrijednost 0.

b) Koristeći zadane inicijalizirane varijable i popunjeno polje na praznoj crti potrebno je napisati naredbu koja poziva funkciju iz a) dijela zadatka.

```
#define MAXR 10
#define MAXS 20
```

```
...
```

```
int mat[MAXR][MAXS];
int redaka, stupaca, x, y, rez;
```

```
... /* inicijalizacija varijabli i punjenje polja */
```

```
rez = _____;
```

Zadatak 3. (20 bodova)

a) Napisati funkciju generiraj zadanu prototipom:

```
void generiraj(char *ulaz, char *rezultat);
```

Funkcija generira znakovni niz rezultat znakovima iz znakovnog niza ulaz. Znakovi iz ulaznog niza odabiru se slučajnim odabirom. Pretpostaviti kako je duljina oba znakovna niza jednaka te su nizovi prije poziva funkcije ispravno inicijalizirani (npr. znakovni niz rezultat sadrži dovoljno mjesta za pohranu znakova).

b) Napisati glavni program u kojem se inicijalizira generator slučajnih brojeva te poziva funkcija generiraj iz a) dijela zadatka. Funkciju je potrebno pozvati 5 puta te na standardni izlaz za svaki poziv funkcije ispisati rezultat. Pri svakom pozivu funkcije kao ulazni niz potrebno je predati konstantni znakovni niz "aeiou".

Primjer ispisa:

1. generirani rezultat je: aaeei

...

5. generirani rezultat je: iaueo

Zadatak 4. (6 bodova)

Koja vrijednost je pohranjena u varijablu tipa float ako je njezin sadržaj:

a) 00000000 00000000 00000000 00000001

Rješenje: _____

b) 10000001 00000000 00000000 00000000

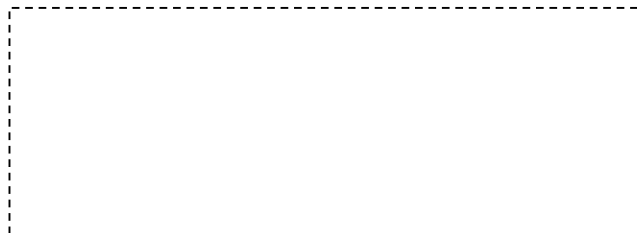
Rješenje: _____

Zadatak 5. (8 bodova)

U naznačeni prostor napisati rezultat izvođenja programa:

```
#include<stdio.h>
#define TDIV(A,B) 3+A/B

int main() {
    printf("%4d,%f,%d\n"
           , 3%2, 4./3
           , TDIV(4 + 4, 3 + 3));
    return 0;
}
```



Zadatak 6. (6 bodova)

Dopuniti kod tako da funkcija mystrcat kopira znakovni niz source na kraj znakovnog niza dest, bez korištenja drugih funkcija iz zaglavlja string.h. Funkcija mystrcat je vlastita implementacija funkcije strcat.

```
char * mystrcat(char *dest, char *source){
    while (*(dest++) != '\0');

    while (*source != _____){
        _____ = *(source++);
    }

    *dest = _____;
    return dest;
}
```

Rješenja i kriterij ocjenjivanja

```
1.
#include <stdio.h>
#include <string.h>

typedef struct zpr {
    int sifra;
    char naziv[95 + 1];
} z_predmet;

void statistika(char *predmet, char *datum) {
    FILE *fispiti, *fpredmeti;
    char t_datum[10 + 1], t_jmbag[4 + 1];
    int t_sifra, t_ocjena;
    int n[5] = { 0 };
    z_predmet temp;
    int sifpred = -1;
    fispiti = fopen("ispiti.txt", "r");
    fpredmeti = fopen("predmeti.bin", "rb");
    while (fread(&temp, sizeof(temp), 1, fpredmeti)) {
        if (!strcmp(temp.naziv, predmet)) {
            sifpred = temp.sifra;
            break;
        }
    }
    while (fscanf(fpredmeti, "%s %s %d %d", t_datum, t_jmbag, t_sifra, t_ocjena) == 4) {
        if (t_sifra == sifpred && !strcmp(t_datum, datum)) {
            n[t_ocjena - 1] ++;
        }
    }
    printf("Broj izislih: %d\n", n[0] + n[1] + n[2] + n[3] + n[4]);
    printf("Izvrstan: %d\nVrlo dobar: %d\nDobar: %dDovoljan: %dNedovoljan: %d"
        , n[4], n[3], n[2], n[1], n[0]);
    fclose(fispiti);
    fclose(fpredmeti);
}
```

2.

a)

```
int lok(int *mat, int r, int s, int maxSt, int x, int y){
    int i, j, cnt;
    float uk_prosjek, lok_prosjek;

    lok_prosjek = uk_prosjek = 0.;
    cnt = 0;
    for (i = 0; i < r; i++){
        for (j = 0; j < s; j++){
            uk_prosjek += *(mat + i*maxSt + j);

            if (i >= (x - 1) && i <= (x + 1) && j >= (y - 1) && j <= (y + 1)){
                lok_prosjek += *(mat + i*maxSt + j);
                cnt++;
            }
        }
    }
    uk_prosjek /= r*s;
    lok_prosjek /= cnt;

    return lok_prosjek < uk_prosjek;
}

//alternativno
int lok(int *mat, int r, int s, int maxSt, int x, int y){
    int i, j, cnt;
    float uk_prosjek, lok_prosjek;

    lok_prosjek = uk_prosjek = 0.;

    for (i = 0; i < r; i++){
        for (j = 0; j < s; j++){
            uk_prosjek += *(mat + i*maxSt + j);
        }
    }
    uk_prosjek /= r*s;

    cnt = 0;
    for (i = x - 1; i <= x + 1 && i < r; i++){
        for (j = y - 1; i >= 0 && j <= y + 1 && j < s; j++){
```

```

        if (j >= 0){
            lok_prosjek += *(mat + i* maxSt + j);
            cnt++;
        }
    }
}
//ili
/*for (i = x - 1; i <= x + 1; i++){
    for (j = y - 1; j <= y + 1; j++){
        if (i >= 0 && j >= 0 && i < r && j < s){
            lok_prosjek += *(mat + i*maxSt + j);
            cnt++;
        }
    }
}*/
lok_prosjek /= cnt;
return lok_prosjek < uk_prosjek;
}

```

b)

```

#define MAXR 10
#define MAXS 20
...
int mat[MAXR][MAXS];
int redaka, stupaca, x, y;
... /* inicijalizacija varijabli i punjenje polja */
res = lok_ispod_prosjeka(&mat[0][0], redaka, stupaca, MAXS, x, y);

```

3.

a)

```

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

#define MAX 10
#define CNT 5

int main(void){
    char rezultat[MAX];
    int i = 0;

    srand(time(NULL));
    for (i = 0; i < CNT; i++){
        generiraj("aeiou", rezultat);
        printf("%d. generirani rezultat je: %s", rezultat);
    }
}

```

b)

```

void generiraj(char *ulaz, char *rezultat){
    int i, len, pos;

    len = strlen(ulaz);
    for (i = 0; i < len; i++){
        pos = rand() % len;
        rezultat[i] = ulaz[pos];
    }
    rezultat[len] = '\0';
}

```

4. a) 00000000 00000000 00000000 00000001

Rješenje: $2^{-149} \sim 1.4E-45$

b) 10000001 00000000 00000000 00000001

Rješenje: $-2^{-125} \sim -2.35E-38$

5. Ispis (prvi red su oznake kolona):

123456789012

1,1.33,11

6.

```
char * strcat(char *dest, char *source){
    while (*dest != '\0');

    while (*source != _____ '\0'){
        _____ *(dest++) = *(source++);
    }

    *dest = _____ '\0';
    return dest;
}
```