

JMBAG		Ime i prezime	
-------	--	---------------	--

Programiranje i programsko inženjerstvo

Ispitni rok

01. srpnja 2014.

Odgovore na 1., 2. i 3. pitanje napišite na svojim papirima i predajte u košuljici.

Zadatak 1. (20 bodova)

Napisati funkciju za generiranje šifre. Prototip funkcije je zadan sa:

```
void generirajSifru(int brSlova, int brZnamenki, char *sifra)
```

Funkcija generira šifru (vraća je preko parametra `sifra`) koja se sastoji od prvih `brSlova` znakova sastavljenih od slučajnih velikih slova engleske abecede (A-Z), nakon čega slijedi `brZnamenki` znakova sastavljenih od slučajnih znamenki (0-9).

Na primjer za zadani broj slova `brSlova` 5 i broja znamenki `brZnamenki` 3 funkcija može vratiti šifru YZABB712. Pretpostaviti da je za šifru rezervirano dovoljno znakova.

Napisati glavni program koji za zadani parametar n generira i ispisuje na zaslon n šifri: prvu sa 0 slova i n slučajnih znamenki, drugu sa 1 slučajnim slovom i $n-1$ slučajnih znamenki, ..., n -tu sa n slučajnih slova i 0 znamenki.

Npr. za dani $n=5$ glavnim programom se može ispisati:

```
76650
C4521
VG018
BHT95
SGSD3
LKTCV
```

Zadatak 2. (25 bodova)

Fotografija (dimenzija $n \times n$ točaka tj. piksela) je u računalu predstavljena cjelobrojnomo matricom dimenzija $n \times n$ s vrijednostima u rasponu od 0 do 255 koje predstavljaju odgovarajuću boju (nijansu sive) piksela. Napisati funkciju koja će sažeti fotografiju ($n \times n$) za 50% po obje dimenzije stvaranjem nove fotografije (dimenzija $(n/2) \times (n/2)$) u kojoj svaki element nastaje računanjem aritmetičke sredine 4 vrijednosti originalne matrice kao što je prikazano na slici. Aritmetička sredina zaokružuje se na manji cijeli broj.

Na primjer, za početnu matricu

101	45	5	6
120	112	10	0
200	210	80	85
79	45	255	255

funkcija vraća matricu:

194	5
133	168

Napisati glavni program u kojem će se učitati kvadratna matrica parnih dimenzija ne većih od 100, pozvati funkcija za sažimanje te ispisati sažeta matrica na zaslon.

Zadatak 3. (20 bodova)

Svaki zapis postojeće neformatirane (binarne) datoteke "uv_indeks.dat" sadrži poštanski broj grada (cijeli broj od 5 znamenki), UV indeks (cijeli broj iz zatvorenog intervala $[0, 10]$) i oznaku razine štetnosti UV zračenja (niz znakova duljine najviše 20 znakova).

Napisati **glavni program** koji će ažurirati zapise datoteke oznakama razine štetnosti na način da za one zapise kojima je UV indeks veći od prosječnog u datoteci postavi razinu štetnosti na "visoka", a za ostale "niska".

Odgovore na 4, 5. i 6. pitanje napišite na za to predviđenom mjestu uz zadatak.

Zadatak 4. (3 boda)

U prostoru označenom pravokutnikom napišite što će se ispisati na ekranu izvođenjem sljedećeg programskog odsječka:

```
int polje[6] = {4, 7, 2};
printf("%d %d %d", polje[1],
        polje[3], polje[6]);
```

Zadatak 5. (8 bodova)

a) (4 boda) Nadopuniti funkciju `najveciZnak`, tako da za zadani niz slova (bez praznina) niz vraća najveće slovo engleske abecede koje se pojavljuje u nizu (najveće slovo engleske abecede je Z to jest z). Funkcija radi neovisno o velikim i malim slovima (Npr N i n je veće od m).

Na primjer za niz "Zagreb" funkcija vraća Z.

(Na jednoj liniji napisati samo jednu naredbu.)

```
char najveciZnak(char niz[], int duljina) {
    char rez; int i;
```

```
    for (i = 1; i < duljina; i++)
```

```
        return rez;
```

```
}
```

b) (4 boda) Nadopuniti glavni program u kojem se uz pomoć funkcije `najveciZnak` (iz a) dijela zadatka) izračunava i ispisuje broj pojavljivanja najvećeg znaka u zadanom nizu znakova.

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

```
int main(void) {
    char niz[30], maxZnak;
    int i = 0, br = 0;
    gets(niz);
    maxZnak = _____;

    while (_____) {

        _____
        i++;
    }
    printf("Broj pojavljivanja najvećeg znaka u nizu: %d", br);
}
```

Zadatak 6. (4 boda)

U prostoru označenom pravokutnikom napisati što će se ispisati na ekranu izvođenjem sljedećeg programskog odsječka:

```
int i;
int brojevi[4] = { 0x10, 010, 10, 0010 };

for (i = 0; i < 4; ++i)
    printf("%d ", brojevi[i]);
```

Rješenja:

Zadatak 1.

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

void generirajSifru(int brSlova, int brZnamenki, char *sifra)
{
    int i, br=0;

    for (i=0; i < brSlova; i++){
        char c = (char) ('A' + rand()%('Z' - 'A' + 1));
        sifra[br] = c;
        br++;
    }

    for (i=0; i < brZnamenki; i++){
        char c = '0' + rand() % 10;
        sifra[br] = c;
        br++;
    }
    sifra[br] = '\0';
}

int main(void) {

    char sifra[10+1];
    int n, i;

    printf("Učitaj n:");
    scanf("%d", &n);

    srand(time(NULL));

    for (i=0; i < n; i++){
        generirajSifru(i, n-i, sifra);
        printf("sifra: %s\n", sifra);
    }

    return 0;
}
```

Zadatak 2.

```
#include <stdio.h>
#define MAX 100

void Sazmi(int *mat1, int n_mat1, int *mat2, int *n_mat2, int max)
{
    int i,j, i_pom = 0, j_pom= 0;

    for (i=0; i<n_mat1; i=i+2){
        for (j=0; j<n_mat1; j=j+2) {
            mat2[i_pom*max + j_pom] = (mat1[i*max + j] +
            mat1[(i+1)*max + j] + mat1[i*max + (j+1)] +
            mat1[(i+1)*max + (j+1)])/4;
            j_pom++;
        }
        i_pom++; j_pom = 0;
    }
    *n_mat2 = n_mat1/2;
}
```

```

int main(void) {

    int i, j, n_mat1, n_mat2;
    int mat1[MAX][MAX], mat2[MAX/2][MAX/2];

    do {
        printf("Unesi n:");
        scanf ("%d", &n_mat1);
    } while (n_mat1<0 || n_mat1>MAX || n_mat1%2);

    for (i=0; i<n_mat1; i++){
        for (j=0; j<n_mat1; j++) {
            printf("mat[%d][%d] = ", i, j);
            scanf ("%d", &mat1[i][j]);
        }
    }

    Sazmi(mat1[0], n_mat1, mat2[0], &n_mat2, MAX);

    for (i=0; i<n_mat2; i++){
        for (j=0; j<n_mat2; j++) {
            printf("%d ", mat2[i][j]);
        }
        printf("\n");
    }
    return 0;
}

```

Zadatak 3.

```

typedef struct z {
    int pbr;
    int uvi;
    char razina[20+1];
} zapis;

int main() {
    FILE *f1;
    zapis z1;
    double suma = 0, prosjek, br = 0;
    f1 = fopen("uv_indeks.dat", "r+b");
    while (fread(&z1, sizeof(zapis), 1, f1) == 1) {
        suma += z1.uvi;
        br++;
    }
    prosjek = suma/br;
    fseek(f1, 0, SEEK_SET);
    while (fread(&z1, sizeof(zapis), 1, f1) == 1) {
        if (z1.uvi > prosjek){
            strcpy(z1.razina, "visoka");
        }
        else{
            strcpy(z1.razina, "niska");
        }
        fseek(f1, -sizeof(zapis), SEEK_CUR);
        fwrite(&z1, sizeof(zapis), 1, f1);
    }
    fclose(f1);
}

```

Zadatak 4.

7 0 ?(nepoznata vrijednost)

Zadatak 5.

(a)

```
char najvećiZnak(char niz[], int duljina) {
    char rez; int i;
    rez = toupper(niz[0]);
    for (i = 1; i < duljina; i++)
        if (toupper(niz[i]) > rez) rez = toupper(niz[i]);
    return rez;
}
```

(b)

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

int main(void) {
    char niz[30], maxZnak;
    int i = 0, br = 0;
    gets(niz);
    maxZnak = najvećiZnak(niz, strlen(niz));
    while (niz[i] != '\0'){
        if (niz[i] == maxZnak) br++;
        i++;
    }
    printf("Broj pojavljivanja najvećeg znaka u nizu: %d", br);
}
```

Zadatak 6.

16 8 10 8