

JMBAG	
Ime i prezime	

**Programiranje i programsko inženjerstvo**  
**Završni ispit**  
**20. siječnja 2010.**

**Odgovore na 1., 2. i 3. pitanje napišite na svojim papirima i predajte u košuljici.**  
**Odgovore na pitanja od 4. do 6. napišite na za to predviđenom mjestu uz zadatke.**  
**Nije dopušteno korištenje globalnih varijabli i naredbe goto.**

**1. (8 bodova)**

- a) Napišite funkciju `maskiraj` koja preko parametara prima matrice `mat` i `maska`. Matrice su istih dimenzija. `mat` sadrži bilo koji cijeli broj, dok `maska` sadrži samo brojeve 0 ili 1. Funkcija treba obaviti maskiranje matrice `mat` matricom `maska`. Maskiranje se obavlja tako da na mjestu  $(i,j)$  matrice `mat` ostaje originalna vrijednost ako se na istom mjestu matrice `maska` nalazi vrijednost 1, inače se vrijednost u matrici `mat` zamjenjuje s nulom.
- b) Napišite glavni program u kojem će se učitati dimenzije matrica (maksimalno 100x150) i zatim uz pomoć generatora pseudoslučajnih brojeva generirati matrice `mat` i `maska` te pozvati funkciju `maskiraj`. Generirani brojevi u matrici `mat` moraju biti iz intervala [100,200], dok matrica `maska` sadrži samo brojeve 0 i 1. Glavni program treba ispisati matricu `mat` nakon maskiranja.

**2. (8 bodova)**

Svaki redak formatirane datoteke s jelovnikom sadrži naziv jela (maksimalno 50 znakova bez praznina) i cijenu jela (realni broj).

- a) Napišite funkciju čiji je prototip
- ```
int najjeftiniji(FILE *dat, char *podniz, char *naziv, float *cijena)
```
- koja će u datoteci s jelovnikom za koju je otvoren tok podataka `dat` pronaći najjeftinije jelo koje u svom nazivu sadrži zadani podniz te naziv tog jela i cijenu vratiti preko argumenata `naziv` i `cijena`. Funkcija preko svog imena vraća 1 ako postoji barem jedno jelo koje sadrži navedeni podniz, a 0 ako u datoteci ne postoji jelo koje sadrži zadani podniz (u tom slučaju povratne vrijednosti za naziv i cijenu nisu definirane).
- Napomena:** Može se pretpostaviti da sva jela imaju različite cijene.
- b) Napišite glavni program u kojem će se s tipkovnice učitati dva niza znakova (ne dulja od 20 znakova), zatim s pomoću funkcije `najjeftiniji` odrediti najjeftinije jelo i njegova cijena u datoteci `jelovnik55.txt` za svaki od unesenih nizova te ispisati odgovarajuća poruka (naziv jela i njegova cijena ili poruka da ne postoji jelo čiji naziv sadrži uneseni niz znakova).

**3. (8 bodova)**

- a) Napišite funkciju `insert` koja **bez korištenja pomoćnog polja** u zadani niz `niz1` **iza znaka** s indeksom člana niza `pozicija` ubacuje niz `niz2`. Funkcija vraća pokazivač na `niz1`. Prototip funkcije je:

```
char *insert (char *niz1, char *niz2, int pozicija);
```

**Primjer:** za učitane nizove `niz1` ABCDE, `niz2` FG i poziciju 2 funkcija `insert` mijenja `niz1` u ABCFGDE.

- b) Napišite glavni program koji će s tipkovnice učitati nizove `niz1` i `niz2` i poziciju (indeks člana niza) **iza koje** `niz2` treba ubaciti u `niz1`. Pozvati funkciju `insert` i na zaslon ispisati niz `niz1`.

**Napomena:** nije potrebno obavljati provjere duljina ulaznih nizova te vrijednost pozicije - pretpostavlja se da će svi uneseni podaci biti takvi da će se funkcija moći ispravno obaviti.

#### 4. (2 boda)

Definirajte macro s parametrima `OROCENJE` koji vraća iznos kapitala nakon `n` godina ukamaćivanja za zadane parametre: `C0` - početni iznos kapitala, `p` - godišnja kamata, `n` - vrijeme ukamaćivanja (u godinama). Iznos kapitala računa se prema formuli:  $C0 \cdot (1 + n \cdot p / 100)$ .

#### 5. (2 boda)

U neformatiranu datoteku `integeri.bin` upisani su cijeli brojevi (`int`). Koristeći prostor nad crtama nadopunite program tako da on pročita prvi broj s početka datoteke, pročitan broj pretvori u negativni te ga zapiše na mjesto zadnjeg cijelog broja u datoteci (prepiše preko trenutno zadnjeg cijelog broja u datoteci). Iznad svake crte dopušteno je napisati samo jednu naredbu.

```
#include <stdio.h>
#include <stdlib.h>
int main() {
    FILE *f;
    int i;
    f = fopen("integeri.bin", _____);
    if (f == NULL) exit(-1);

    _____;
    i = -i;

    _____;

    _____;
    fclose(f);
    return 0;
}
```

#### 6. (2 boda)

Koristeći prostor nad crtama, nadopunite dijelove programa (definiranje novog tipa, definiranje varijable tog tipa) kako bi programski odsječak bio ispravan.

```
#include <stdio.h>
int main() {

    _____ struct {

        int _____;

    } _____;

    mojtip _____;
    scanf("%d", &x.a);
    ...
}
```

Rješenja:

1.

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

#define MAXSTUP 150
#define MAXRED 100

void maskiraj(int *mat, int *maska, int r, int s, int maxstup){
    int i,j;
    for(i=0;i<r;i++){
        for(j=0;j<s;j++){
            mat[i*maxstup+j]*=maska[i*maxstup+j];
        }
    }
}

int main(){
    int r,s,i,j;
    int mat[MAXRED][MAXSTUP], maska[MAXRED][MAXSTUP];
    srand((unsigned)time(NULL));

    do{
        scanf("%d %d",&r,&s);
    }while(r<1 || r>MAXRED || s<1 || s>MAXSTUP);

    for(i=0;i<r;i++){
        for(j=0;j<s;j++){
            mat[i][j]=rand()%(200-100+1)+100;
            maska[i][j]=rand()%2;
        }
    }

    maskiraj(&mat[0][0],&maska[0][0],r,s,MAXSTUP);

    for(i=0;i<r;i++){
        printf("\n");
        for(j=0;j<s;j++){
            printf("%3d ",mat[i][j]);
        }
    }
    return 0;
}
```

2.

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

int najjeftiniji(FILE *dat, char *podniz, char *naziv, float *cijena){
    float cijena_jela; /*procitana cijena u pojedinom koraku*/
    char naziv_jela[51]; /*procitani naziv u pojedinom koraku*/

    *cijena = 0;
    fseek(dat, 0, SEEK_SET);
    while(fscanf(dat, "%s %f", naziv_jela, &cijena_jela) != EOF){
        if (strstr(naziv_jela, podniz) != NULL){
            if (*cijena == 0 || cijena_jela < *cijena){
                strcpy(naziv, naziv_jela);
                *cijena = cijena_jela;
            }
        }
    }
    return *cijena > 0;
}

int main(){
    FILE *f;
    char podniz1[21], podniz2[21], naziv[51];
    float cijena;
    f = fopen("jelovnik55.txt", "r");
    if (f == NULL){
        printf("Datoteku nije moguće otvoriti\n");
        exit(1);
    }

    printf("Unesite dva podniza\n");
    scanf("%s %s", podniz1, podniz2);

    if (najjeftiniji(f, podniz1, naziv, &cijena))
        printf("%s %f\n", naziv, cijena);
    else
        printf("Ne postoji jelo koje sadrži podniz %s\n", podniz1);

    if (najjeftiniji(f, podniz2, naziv, &cijena))
        printf("%s %f\n", naziv, cijena);
    else
        printf("Ne postoji jelo koje sadrži podniz %s\n", podniz2);

    fclose(f);
    return 0;
}
```

3.

```
char * insert ( char * niz1, char * niz2, int pozicija)
{
    int duljina1=strlen(niz1);
    int duljina2= strlen(niz2);
    int i;
    for(i=duljina1; i>pozicija; i--){
        niz1[i+duljina2]=niz1[i];
    }
    strncpy(niz1+pozicija +1, niz2, duljina2);
    return niz1;
}

int main(){
    char niz1[110], char niz2[10];
    int pozicija;
    printf("Ucitajte niz1: ");
    scanf("%s", niz1);
    printf("\nUcitajte niz2: ");
    scanf("%s", niz2);
    printf("\nUcitajte poziciju: ");
    scanf("%d", &pozicija);
    insert(niz1, niz2, pozicija);
    printf("%s", niz1);
    return 0;
}
```

4.

```
#define OROCENJE(C0,p,n) ((C0)*(1+(n)*(p)/100))
```

5.

```
#include <stdio.h>
#include <stdlib.h>
int main() {
    FILE *f;
    int i;
    f = fopen("integeri.bin", "r+b");
    if (f == NULL) exit(-1);

    fread(&i, sizeof (i), 1, f);
    i = -i;

    fseek(f, -1*sizeof(int), SEEK END);

    fwrite(&i, sizeof (i), 1, f);

    fclose(f);
    return 0;
}
```

6.

```
#include <stdio.h>
int main() {

    typedef struct {

        int       a      ;

    } mojtip;

    mojtip       x      ;
    scanf("%d", &x.a);
}
```