

Programiranje i programsko inženjerstvo – 2. Ispitni rok

2. srpnja 2013.

Nije dopušteno korištenje globalnih i statičkih varijabli te naredbe **goto**. Ispit nosi maksimalno 80 bodova, a prag za prolaz pismenog ispita je **40** bodova. Ovaj obrazac morate predati.

Zadatak 1. (25 bodova)

Tekstualna datoteka **ispitniRok_Uvidi.txt** sadrži zapise o rezultatima studenata na ispitu iz *Programiranja i programskog inženjerstva* nakon uvida. Svaki se redak u datoteci sastoji od šifre studenta (cijeli broj) te bodova iz 6 zadataka (realni brojevi s jednom decimalom). Podatci u pojedinom retku datoteke su međusobno odvojeni jednom prazninom, npr.

1234 1.0 10.5 2.5 13.5 15.0 10.5

gdje je 1234 šifra studenta, 1.0 broj bodova koje je student ostvario u 1. zadatku, 10.5 broj bodova koje je student ostvario u 2. zadatku, itd.

U direktnoj binarnoj datoteci **studentiProg.bin** nalaze se zapisi o studentima koji su pisali ispit iz predmeta *Programiranje i programsko inženjerstvo*. Zapis se sastoji od šifre studenta (cijeli broj), imena studenta (najviše 30 znakova), prezimena studenta (najviše 50 znakova) i broja bodova ostvarenog na ispitu (realni broj). Redni broj zapisa u datoteci odgovara šifri studenta.

- (5 bodova)** Napisati funkciju `zbroyiBodove` koja će za zadani broj zadataka i jednodimenzionalno polje s bodovima po zadacima izračunati sumu bodova. Funkcija mora vratiti sumu bodova ako je ona veća ili jednaka pragu prolaznosti od 30 bodova. Ako je suma bodova manja od praga prolaznosti funkcija mora vratiti 0.
- (20 bodova)** Napisati program koji će u datoteci **studentiProg.bin** ažurirati broj bodova za studente kojima je nakon uvida promijenjen ostvareni broj bodova. Pretpostaviti da su svi studenti iz **ispitniRok_Uvidi.txt** već u **studentiProg.bin**. Broj bodova se određuje pozivanjem funkcije `zbroyiBodove`.

Napomena: nije potrebno provjeravati uspješnost otvaranja datoteka te uspješnost pozicioniranja u datoteci.

Zadatak 2. (10 bodova)

- Prikažite, u heksadekadskom obliku, realni broj -10,25 prema IEEE 754 standardu u jednostrukoj preciznosti.

- Prikažite, u heksadekadskom obliku, sadržaj registra u kojem je prema IEEE 754 standardu u jednostrukoj preciznosti pohranjen broj $2_{10} * 2^{-128}$

- Dekadski broj je pohranjen prema IEEE 754 standardu u jednostrukoj preciznosti. Heksadekadska vrijednost sadržaja registra je 3F800000. Koji je to dekadski broj?

Zadatak 3. (5 bodova)

Nadopuniti funkciju `zamijeniZnakom`, tako da se u zadanom nizu `niz` svako pojavljivanje znaka `z` zamjenjuje slučajnim malim slovom engleske abecede.

Pretpostaviti da će generator slučajnih brojeva biti inicijaliziran prije poziva funkcije `zamijeniZnakom`.

Nije dopušteno koristiti dodatne programske petlje.

```
#include <string.h>
#include <stdlib.h>
...
int zamijeniZnakom (char *niz, char z) {
    while ( _____ ) {

        *niz = _____;
    }
    return 0;
}
```

Zadatak 4. (40 bodova)

Ispit se sastoji od pitanja s ponuđenim odgovorima. Odgovori studenata pohranjeni su u dvodimenzionalnom polju znakova, za svakog studenta po jedan redak s njegovim odgovorima (u stupcu sa indeksom 0 nalazi se odgovor na prvo pitanje itd.). Točni odgovori po pitanjima pohranjeni su u jednodimenzionalnom polju znakova koje za svako pitanje sadrži točan odgovor. Redni broj pitanja odgovara indeksu u polju uvećanom za 1 (točan odgovor na prvo pitanje nalazi se na indeksu 0 itd.).

- a) **(20 bodova)** Napisati funkciju `ocijjeni` koja će kao parametre primiti odgovore studenata i točne odgovore po pitanjima te u pozivajući program vratiti ostvareni broj bodova za sve studente kao jednodimenzionalno polje realnih brojeva.
- U odgovorima studenta može se naći i znak '-' koji označava da student nije odabrao niti jedan odgovor. Ako je odgovor točan student dobiva 1 bod, ako odgovor nije točan student dobiva -0,5 bodova a 0 bodova dobiva ako nije odabrao niti jedan odgovor.
- b) **(20 bodova)** Napisati program u kojem će se iz tekstualne datoteke **rezultati.txt** pročitati odgovori studenata na ispitu sa 6 pitanja i 4 moguća odgovora i pohraniti u dvodimenzionalno polje znakova. Možete pretpostaviti da je maksimalan broj studenata 100. U datoteci se nalaze odgovori studenata (u svakom retku odgovori jednog studenta) koji su međusobno odvojeni jednom prazninom. Npr. ako se u datoteci nalaze odgovori za 2 studenta datoteka će imati sljedeći sadržaj:

A D C A B C
D - B A - C

Uz definirane točne odgovore za 6 pitanja

D B C A A C

potrebno je pozvati funkciju `ocijjeni` i njen rezultat ispisati na ekranu tako da se broj bodova za svakog studenta ispisuje u novom retku. Za gore navedene primjere odgovora studenata ispis bi bio:

1.5
2.5

jer prvi student ima 3 točna odgovora i 3 netočna odgovora. Drugi student ima 3 točna odgovora, 1 netočan odgovor i 2 neodgovorena pitanja.

Napomena: nije potrebno provjeravati uspješnost otvaranja datoteka te uspješnost pozicioniranja u datoteci

Rješenja:

Zadatak 1.

```
#include <stdio.h>
#define BRZAD 6
```

```
typedef struct s {
    int sifStudent;
    char imeStudent[30 + 1];
    char prezStudent[50 + 1];
    float bodovi;
} zapis;
```

a)

```
float zbrojiBodove(int brZadataka, float bodovi[]){
    int i = 0;
    float suma = 0;
    for(i = 0; i < brZadataka; i++){
        suma += bodovi[i];
    }
    if(suma >= 30){
        return suma;
    } else{
        return 0;
    }
}
```

b)

```
int main() {
    FILE *fIspitRok = NULL, *fStudent = NULL;
    float bodovi[BRZAD] = {0.};
    int sifStudent;
    zapis student;
    float bodoviStud = 0;
    int ocjena, i = 0;

    fIspitRok = fopen("ispitniRok_Uvidi.txt", "r");
    fStudent = fopen("studentiProg.bin", "r+b");

    while (fscanf(fIspitRok, "%d %f %f %f %f %f", &sifStudent, &bodovi[0], &bodovi[1], &bodovi[2],
        &bodovi[3], &bodovi[4], &bodovi[5]) == BRZAD + 1) {
        bodoviStud = zbrojiBodove(BRZAD, bodovi);

        fseek(fStudent, (sifStudent - 1) * sizeof(zapis), SEEK_SET);
        fread(&student, sizeof(zapis), 1, fStudent);

        if(bodoviStud != student.bodovi){
            student.bodovi = bodoviStud;
            fseek(fStudent, sizeof(zapis) * (-1L), SEEK_CUR);
            fwrite(&student, sizeof(zapis), 1, fStudent);
        }
    }

    fclose(fIspitRok);
    fclose(fStudent);
    return 0;
}
```

Zadatak 2.

a) $-10.25_{10} = 1010.01_2 = 1.01001 \cdot 2^3$

$P=1$

$BE = 3, K = 127 + 3 = 130_{10} = 10000010_2$

$M = 010010000000000000000000$

Rješenje:

$11000001001001000000000000000000_{(2)}$

$C1240000_{(16)}$

b) Kada je $K = 0$ i postoje binarne frakcije u mantisi tada je to denormalizirani broj, tj. više ne postoji skriveni bit.

$2_{10} \cdot 2^{-128} = 10_2 \cdot 2^{-128} = 0.1_2 \cdot 2^{-126}$

$P=0$

$K=0$

$M=100000000000000000000000$

Rješenje:

$00000000010000000000000000000000_{(2)}$

$00400000_{(16)}$

c) Sadržaj registra je: $00111111100000000000000000000000_{(2)}$

$P=0$

$K=127, BE=0$

$M=0$

Rješenje:

1

Zadatak 3.

```
int zamijeniZnakom (char *niz, char z) {  
    while (niz = strchr(niz, z)) {  
        *niz = rand() % 26 + 'a';  
    }  
    return 0;  
}
```

Zadatak 4.

a)

```
void ocijeni(char* odgStud, int nStudenata, int maxZad, int nZad, char* tocOdg, float *rez){
    int i, j;
    float bodoviStud = 0;
    char odgovor = '\0';
    /*Prodji kroz sve studente*/
    for(i = 0; i < nStudenata; i++){
        bodoviStud = 0;
        /*Prodji kroz sve odgovore studenta*/
        for(j = 0; j < nZad; j++){
            /*Pohrani odgovor studenta u privremenu varijablu*/
            odgovor = odgStud[i * maxZad + j];
            if(odgovor == tocOdg[j]){
                /*Ako je odgovor tocan +1 bod*/
                bodoviStud += 1;
            }else if(odgovor != tocOdg[j] && odgovor != '-'){
                /*Ako je odgovor nije tocan niti je neodgovoren -0.5 bodova*/
                bodoviStud -= 0.5f;
            }
        }
        /*Pohrani ostvareni rezultat u polje koje se vraca u pozivajuci program*/
        rez[i] = bodoviStud;
    }
}
```

b)

```
#define MAXSTUD 100
#define BRZAD 6
#define BRODG 4

int main(){
    FILE *fRezultati = NULL;
    char odgS[MAXSTUD][BRZAD] = {'\0'};
    char tocOdg[BRZAD] = {'D', 'B', 'C', 'A', 'A', 'C'};
    float rezultati[MAXSTUD] = {0.0f};
    int brStud = 0, i = 0;
    char enter;

    /*Procitaj odgovore studenata*/
    fRezultati = fopen("rezultati.txt", "r");
    while(fscanf(fRezultati, "%c %c %c %c %c %c %c %c %c %c",
        &odgS[brStud][0], &odgS[brStud][1], &odgS[brStud][2], &odgS[brStud][3],
        &odgS[brStud][4], &odgS[brStud][5], &enter) == BRZAD + 1)
    {
        brStud++;
    }
    fclose(fRezultati);

    /*Ocijeni studente*/
    ocijeni(&odgS[0][0], brStud, BRZAD, BRZAD, tocOdg, rezultati);

    /*Ispisi rezultate*/
    for(i = 0; i < brStud; i++){
        printf("Ostvareni rezultat %d. studenta je %g\n", i+1, rezultati[i]);
    }

    return 0;
}
```