

## Završni ispit iz Programiranja i programskog inženjerstva 27. lipnja 2006.

Napomene za sve zadatke:

- Nije dopušteno korištenje goto naredbe, te statičkih i globalnih varijabli.
- Studentima na pitanja ne smiju odgovarati asistenti koji ih čuvaju u učionici. Na pitanja će odgovarati isključivo asistenti koji za vrijeme ispita obilaze prostorije

1. (7 bodova) Napisati **funkciju** kojoj se zadaju niz znakova `tekst` i znak `c`. Funkcija mora svaki znak `c` u nizu `tekst` promijeniti u znak 'X'. Funkcija u pozivajući program vraća pokazivač na posljednji znak koji je promijenjen ili NULL pokazivač ako nije promijenjen niti jedan znak. Za traženje znaka unutar niza **obavezno** koristiti odgovarajuću funkciju iz programske biblioteke. Pri promjeni znaka **obavezno** koristiti pokazivač na pronađeni znak.

*Primjeri:* za zadani niz "Sasvim kasno" i znak 's', funkcija vraća pokazivač na 10-ti znak niza, a zadani niz mijenja u "SaXvim kaXno". Za zadani niz "Sasvim kasno" i znak 'R', funkcija vraća NULL pokazivač i ne mijenja zadani niz.

2. (7 bodova) Napisati **program** koji pomoću funkcije puts iz programske biblioteke na zaslon ispisuje niz duljine 8 znakova. Znakovi niza odabiru se na slučajan način među velikim slovima abecede ('A' - 'Z'). Isto slovo se u nizu smije pojaviti više puta.

*Primjer ispisa na zaslon:*

JUTASUTZ
----------

3. (8 bodova) Napisati **funkciju** koja sve parne elemente zadanog dvodimenzionalnog cjelobrojnog polja postavlja na vrijednost 0, neparne elemente na vrijednost 1, te u pozivajući program vraća koliko je u polju bilo parnih, a koliko neparnih elemenata (funkcija vraća dvije vrijednosti). Funkcija mora biti napisana tako da ispravno radi s dvodimenzionalnim cjelobrojnim poljima s bilo kojim brojem redaka i stupaca.

*Primjer 1:* Funkcija će polje

1	0
-1	-3
4	5
6	7

promijeniti u

1	0
1	1
0	1
0	1

te vratiti vrijednosti 3 i 5.

*Primjer2:* Funkcija će polje

1	-2	0
4	3	2
5	6	8

promijeniti u

1	0	0
0	1	0
1	0	0

te vratiti vrijednosti 6 i 3.

Također, napisati glavni program (tj. funkciju `main`) u kojem treba jednom naredbom definirati i inicijalizirati polje na vrijednosti iz *Primjera 1*. Zatim pozvati funkciju te na zaslon ispisati broj parnih i neparnih članova polja.

4. (8 bodova) Svaki zapis postojeće direktne neformatirane (binarne) datoteke "stud.bin" sadrži podatke o jednom studentu: matični broj studenta (int), ime i prezime studenta (niz znakova duljine 40+1), ukupni broj bodova koje je stekao do završnog ispita (float). Redni broj zapisa datoteke "stud.bin" odgovara matičnom broju studenta.

Svaki zapis postojeće slijedne formatirane (tekstualne) datoteke "ispit.txt" sadrži matični broj studenta (cijeli broj) i broj bodova koje je dobio na završnom ispitu (realni broj s jednom znamenkom iza decimalne točke).

*Primjer sadržaja datoteke "ispit.txt":*

452	12.5
32	9.2
617	28.0

Napisati **program** koji će broj bodova svakog studenta u datoteci "stud.bin" **uvećati** za broj bodova koje je student dobio na završnom ispitu. U programu **nije potrebno** ispitivati uspješnost obavljanja operacija nad datotekama (npr. je li uspjelo otvaranje, čitanje, pisanje, itd.).

1.

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

char *zamijeni (char *tekst, char c) {
    char *kazaljka, *zadnjaKazaljka = NULL;
    while ((kazaljka = strchr(tekst, c)) != NULL) {
        zadnjaKazaljka = kazaljka;
        *kazaljka = 'X';
    }
    return zadnjaKazaljka;
}
```

2.

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

int main () {
    int i;
    char niz[8+1];
    srand ((unsigned) time(NULL));
    for (i = 0; i < 8; i++)
        niz[i] = (float) rand() / (RAND_MAX+1) * ('Z'-'A'+1) + 'A';
        // niz[i] = rand() % ('Z' - 'A' + 1) + 'A';
    niz[8] = '\0';
    puts(niz);
    return 0;
}
```

3.

```
#include <stdio.h>

void parnepar(int *polje, int maxred, int m, int n, int *parnih, int *neparnih) {
    int i, j;
    *parnih = *neparnih = 0;
    for (i = 0; i < m; i++)
        for (j = 0; j < n; j++)
            if (*(polje + i*maxred + j) % 2 == 0) {
                *(polje + i*maxred + j) = 0;
                (*parnih)++;
            }
            else {
                *(polje + i*maxred + j) = 1;
                (*neparnih)++;
            }
}

int main() {
    int polje[4][2] = {{1, 0}, {-1, -3}, {4, 5}, {6, 7}};
    int par, nepar;
    parnepar(&polje[0][0], 2, 4, 2, &par, &nepar);
    printf("%d %d\n", par, nepar);
    return 0;
}
```

4.

```
#include <stdio.h>

int main () {
    FILE *studTok, *ispitTok;
    int mbr;
    float brbod;
    struct {
        int matbrStud;
        char prezIme[20+1];
        float ukupBodova;
    } stud;

    studTok = fopen("stud.bin", "r+b");
    ispitTok = fopen("ispit.txt", "r");

    while (fscanf(ispitTok, "%d %f", &mbr, &brbod) == 2) {
        fseek(studTok, (mbr-1)*sizeof(stud), SEEK_SET);
        fread(&stud, sizeof(stud), 1, studTok);
        stud.ukupBodova += brbod;
        fseek(studTok, -1L*sizeof(stud), SEEK_CUR);
        fwrite(&stud, sizeof(stud), 1, studTok);
    }
    fclose(studTok);
    fclose(ispitTok);
    return 0;
}
```