

**Ponovljeni završni ispit iz Programiranja i programskog inženjerstva  
30.1.2007.**

**Napomene uz sve zadatke:**

- nije dopušteno korištenje goto naredbe te statičkih i globalnih varijabli

**1. (7 bodova)**

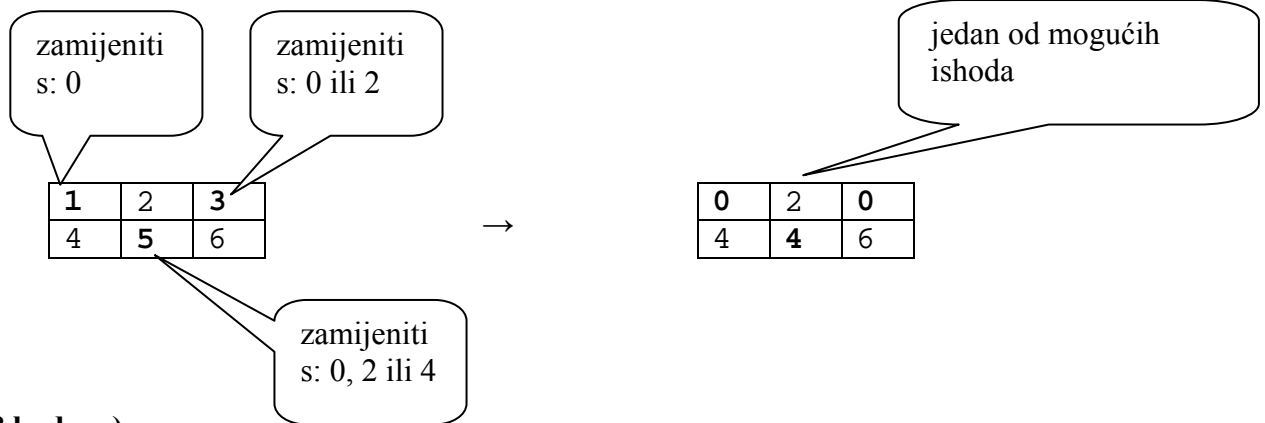
Napisati funkciju koja za zadano jednodimenzionalno cjelobrojno polje u pozivajući program vraća aritmetičku sredinu i sumu članova polja.

Napisati glavni program u kojem je potrebno s tipkovnice uz kontrolu učitati broj elemenata polja (mora biti u intervalu [5,100]) i zatim elemente polja. Koristeći zadanu funkciju ispisati aritmetičku sredinu i sumu članova polja.

**2. (7 bodova)**

Napisati funkciju koja će u zadanoj cjelobrojnoj matrici proizvoljnih dimenzija zamijeniti svaki neparan broj sa slučajno odabranim nenegativnim parnim brojem koji je manji od tog neparnog broja. Matrica sadrži samo nenegativne brojeve u intervalu [0, 20000].

Npr:



**3. (8 bodova)**

Napisati program koji će na temelju **sljedne formatirane** datoteke

ispiti.txt napraviti novu **direktnu neformatiranu** datoteku

polozeniispiti.dat. Datoteka ispiti.txt sadrži podatke o ispitima iz predmeta Matematika na roku 21.1.2007. i u svakom retku sadrži matični broj studenta (cijeli broj od 5 znamenki), ime (najviše 20 znakova bez razmaka), prezime (najviše 30 znakova bez razmaka) i ocjenu (cijeli broj).

U datoteku polozeniispiti.dat treba upisati podatke o **studentima koji su položili ispit.**

Jedan zapis datoteke polozeniispiti.dat treba sadržavati matični broj studenta (int), ime i prezime (51+1 znakova) i ocjenu (int). Primijetiti da je potrebno spojiti ime i prezime u jedan znakovni niz (prilikom spajanja dodati razmak između imena i prezimena).

Matični brojevi studenata su u intervalu [10000, 50000], a redni broj zapisa u datoteci polozeniispiti.dat se određuje na temelju matičnog broja. Kako bi se izbjegla nepotrebna praznina na početku datoteke potrebno je podatke o studentu s matičnim brojem 10000 upisati na prvu poziciju u datoteci, 10001 na drugu, itd.

Nije potrebno provjeravati da li je uspjelo otvaranje datoteka.

**4. (8 bodova)**

Napisati funkciju čiji je prototip:

```
int izbaci(char *tekst, char *podniz)
```

koja će zadani znakovni niz tekst promijeniti tako da iz njega izbaci svaku pojavu zadanog niza podniz osim ako se podniz nalazi na početku ili kraju znakovnog niza tekst. Obavezno koristiti funkciju strstr. Funkcija vraća broj izbačenih podnizova.

ispiti.txt
10012 Ivo Kolar 1
10015 Pero Novak 2
10012 Ivo Kolar 5
10001 Ivan Horvat 3
...

Npr: char s[100] = "AAA bb aaa cc AAA dd AAA"; printf("%d ", izbaci(s, "AAA")); printf("%s", s);	Npr. char s[100] = "bb AAA cc AAA dd"; printf("%d ", izbaci(s, "AAA")); printf("%s", s);
na zaslone će ispisati: 1 AAA bb aaa cc dd AAA	na zaslone će ispisati: 2 bb cc dd

## RJEŠENJA

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

//-----
void arSredSuma(int *niz, int br, int *suma, float *arsred){
    int i=0;
    int s = 0;
    for (i = 0; i < br ; ++i){
        s += niz[i];
    }
    *suma = s;
    *arsred = (float) s / br; // ne treba provjeravati div0
}

int main1(){
    int polje[100];
    int br, i, suma;
    float arsred;
    do {
        scanf("%d", &br);
    } while (br<5 || br>100);
    for (i= 0; i < br; ++i){
        scanf("%d", &polje[i]);
    }
    arSredSuma(polje, br, &suma, &arsred);
    printf("suma=%d arsred=%f", suma, arsred);
    return 0;
}

//-----

void promijeni(int *m, int maxstu, int r, int s) {
    int i, j;
    srand((unsigned)time(NULL));
    for (i = 0; i < r; i++)
        for (j = 0; j < s; j++)
            if (m[i*maxstu+j]%2 == 1){
                m[i*maxstu+j] = 2 * ( rand()%(m[i*maxstu+j]/2-0+1) + 0 );
            }
}

int main2(){
    int i, j;
    int m[10][10] = {{1, 2, 3}, {4, 5, 6}};
    promijeni(&m[0][0], 10, 2, 3);
    for (i = 0; i < 2; i++){
        for (j = 0; j < 3; j++)
            printf("%2d", m[i][j]);
        printf("\n");
    }
    return 0;
}

//-----

int main3(){
    struct {
        int mbr;
        char ip[51+1];
        int ocjena;
    } stud;
```

```

char prezime[30+1];
FILE *ff, *fnf;
ff = fopen("ispiti.txt", "r");
fnf = fopen("polozeniispiti.dat", "wb");
while (fscanf(ff, "%d %s %s %d", &stud.mbr, stud.ip, prezime, &stud.ocjena)
== 4){
    if (stud.ocjena>1){
        strcat(stud.ip, " ");
        strcat(stud.ip, prezime);
        fseek(fnf, (stud.mbr - 10000L) * sizeof(stud), SEEK_SET);
        fwrite(&stud, sizeof(stud), 1, fnf);
    }
}
return 0;
}

//-----

int izbaci(char *niz, char *podniz){
    int br = 0;
    int d = strlen(podniz);
    niz = niz + 1; //
    while ( (niz = strstr(niz, podniz)) != NULL
        && strlen(niz)>d){
        strcpy (niz, niz + d);
        ++br;
    }
    return br;
}

int main() {
    char s[100] = "AAAA bAAAb aaa cc AAA dd AAA";
    printf("%d ", izbaci(s, "AAA"));
    printf("%s", s);
    return 0;
}

```