

JMBAG	
Ime i prezime	

Programiranje i programsko inženjerstvo

Završni ispit - 4. veljače 2016.

Rješenja 1, 2, 3 i 4. zadatka napisati na vlastitim papirima i predati ih u košuljici.
Rješenje 5. i 6. zadatka napisati na za to predviđeno mjesto uz zadatak.

1. (8 bodova)

Napisati definiciju funkcije koja u zadanom dvodimenzijском polju znakova, s proizvoljnim brojem redaka i stupaca, popunjava kvadratnu matricu zadanog reda. Funkciji se, pored argumenata koji opisuju polje, zadaju i koordinate stožernog elementa na temelju kojeg funkcija popunjava matricu na sljedeći način:

- U zadani stožerni element postavlja znak 'X'
- U elemente gore lijevo i dolje desno od stožernog elementa postavlja znak '#'
- U sve ostale elemente postavlja znak '-'

Primjeri:

Red matrice 8 i koordinate stožernog elementa 5, 4	Red matrice 6 i koordinate stožernog elementa 2, 4	Red matrice 7 i koordinate stožernog elementa 6, 6
<pre> # # # # - - - - # # # # - - - - # # # # - - - - # # # # - - - - # # # # - - - - - - - - x - - - - - - - - # # # - - - - - # # # </pre>	<pre> # # # # - - # # # # - - - - - - x - - - - - - # - - - - - # - - - - - # </pre>	<pre> # # # # # # - # # # # # # - # # # # # # - # # # # # # - # # # # # # - # # # # # # - # # # # # # - - - - - - x </pre>

2. (10 bodova)

- (a) Napisati definiciju funkcije zamjenaZnakova koja zamjenjuje vrijednosti u dvije zadane varijable tipa *char*.
- (b) Napisati definiciju funkcije traziSamoglas koja za zadani niz znakova vraća pokazivač na prvi samoglasnik u nizu.
- (c) Napisati glavni program koji s tipkovnice učitava niz znakova ne dulji od 80 znakova. Pozivima funkcije traziSamoglas pronaći prvi i drugi samoglasnik u učitanoj nizu, te ih zamijeniti pomoću funkcije zamjenaZnakova. Na slici su prikazana četiri primjera izvođenja programa.

```

Upisite niz: To jE niz!
Rezultat je: TE jo niz!

Upisite niz: isti su prvi.
Rezultat je: isti su prvi.

Upisite niz: Tukj bz!
Nema dovoljno samoglasnika

Upisite niz: Tkjbz!
Nema dovoljno samoglasnika

```

3. (10 bodova)

Svaki zapis postojeće binarne datoteke porezi.bin sadrži šifru osobe (*int*), ime i prezime (niz znakova duljine 20+1), dohodak ostvaren u 2015. godini (*float*) i porez razrezan dotičnoj osobi za 2015. godinu (*float*). Redni broj zapisa u datoteci odgovara šifri osobe.

Postojeća tekstualna datoteka troskovi.txt sadrži podatke o troškovima liječenja koje su osobe imale tijekom 2015. godine. Svaki zapis sadrži šifru osobe, jedinstveni identifikator računa i plaćeni iznos. Za svaku šifru osobe iz datoteke postoji odgovarajući zapis u datoteci porezi.bin.

Primjer sadržaja datoteke porezi.bin

```

1      Marko Matic 30000.00 2350.00
...
5      Ivo Ivic    20000.00 1550.00
6      Jura Juric  10000.00  750.00
...
1045   Pero Peric  8000.00  550.00

```

Primjer sadržaja datoteke troskovi.txt

```

1045 85525-2fd58-54a24 2020.50
6 232aa-14b21-72d16 450.20
1045 67d33-82f11-1921a 1250.00
...

```

Napisati program kojim će se svim osobama čiji je dohodak manji od prosječnog dohotka svih osoba koje se nalaze u datoteci porezi.bin, porez umanjiti za sve troškove liječenja koje su imali tijekom 2015. godine. Pri tome se porez nikome ne smije smanjiti na iznos manji od nule. Nije potrebno provjeravati uspješnost obavljanja operacija nad datotekama.

4. (8 bodova)

- (a) Napisati definiciju funkcije čiji je prototip `void umetniClan(float *polje, int n, float x)` i koja u zadano jednodimenzijsko realno polje koje sadrži `n` silazno poredanih brojeva umeće element s vrijednošću `x` tako da poredak u polju ostane sačuvan. Funkcija ne mijenja polje ako se pokušava umetnuti element čija je vrijednost manja od u tom trenutku najmanje vrijednosti u polju.
- (b) Napisati glavni program u kojem se definira jednodimenzijsko polje realnih brojeva veličine 5, članovi se inicijaliziraju na nule, a zatim se u polje uzastopnim pozivanjem funkcije `umetniClan` pokušava umetnuti 10 pseudo-slučajnih realnih brojeva s 2 decimale u rasponu od [0, 9.99]. Nakon svakog poziva funkcije na zaslon ispisati koji se pseudo-slučajni broj pokušao umetnuti u polje i kakvim je sadržajem polja poziv funkcije rezultirao.

Primjer izvršavanja programa:

Ubacujem 2.19:	2.19	0.00	0.00	0.00	0.00
Ubacujem 5.85:	5.85	2.19	0.00	0.00	0.00
Ubacujem 8.52:	8.52	5.85	2.19	0.00	0.00
Ubacujem 7.08:	8.52	7.08	5.85	2.19	0.00
Ubacujem 4.62:	8.52	7.08	5.85	4.62	2.19
Ubacujem 0.60:	8.52	7.08	5.85	4.62	2.19
Ubacujem 3.66:	8.52	7.08	5.85	4.62	3.66
Ubacujem 0.99:	8.52	7.08	5.85	4.62	3.66
Ubacujem 4.73:	8.52	7.08	5.85	4.73	4.62
Ubacujem 9.50:	9.50	8.52	7.08	5.85	4.73

5. (2 boda)

U cjelobrojnoj (int) varijabli datum je pohranjen u komprimiranom obliku na sljedeći način:

ne koristi se														godina							mjesec				dan					
31													16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	

Godina se izračunava dodavanjem 1980 na broj predstavljen bitovima 9-15. Na primjer sljedeći sadržaj predstavlja datum iz 2016. godine (jer bitovi 9-15 sadrže broj 36, $36+1980=2016$).

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	1	0	1	1	1	1	0	1	0
0				0				0				0				4				9				7				A			

Nadopuniti sljedeći program tako da na zaslon ispiše godinu za datum pohranjen u varijablu datum:

```
#include <stdio.h>
int main (void) {
    int datum = 0x497A;

    printf("Godina: %d\n", _____);

    return 0;
}
```

6. (2 boda)

Nadopuniti sljedeći program tako da se u drugi element polja studenti s tipkovnice učitaju matični broj, prezime (u kojem sigurno nema bjelina) i dvije ocjene.

```
#include <stdio.h>
int main (void) {
    struct {
        short int mbr;
        char prezime[20+1];
        int ocjene[2];
    } studenti[100];

    scanf(_____,
        _____);

    return 0;
}
```

Rješenja

1.

```
void genmat(char *m, int red, int maxstu, int cx, int cy) {
    int i, j;
    for (i = 0; i < red; ++i)
        for (j = 0; j < red; ++j)
            if ((i < cx && j < cy) || (i > cx && j > cy))
                m[i * maxstu + j] = '#';
            else
                m[i * maxstu + j] = '-';
    m[cx * maxstu + cy] = 'X';
    return;
}
```

2.

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

void zamijeni(char *c1, char *c2) {
    char cuvaj = *c1;
    *c1 = *c2;
    *c2 = cuvaj;
    return;
}

char *traziSamoglas(char *niz) {
    return strpbrk(niz, "aeiouAEIOU");
}

int main (void) {
    char niz[80+1];
    char *samoglas1, *samoglas2;
    printf("Upisite niz: ");
    gets(niz);
    samoglas1 = traziSamoglas(niz);
    if (samoglas1 != NULL) {
        samoglas2 = traziSamoglas(samoglas1 + 1);
    }
    if (samoglas1 != NULL && samoglas2 != NULL) {
        zamijeni(samoglas1, samoglas2);
        printf("Rezultat je: %s", niz);
    }
    else {
        printf("Nema dovoljno samoglasnika.");
    }
    return 0;
}
```

3.

```
#include <stdio.h>

int main (void) {
    struct osoba_s {
        int sifOsoba;
        char imePrez[20+1];
        float dohodak;
        float porez;
    } osoba;
    FILE *troskovi, *porezi;
    int sifOsoba, rbrZapis = 0, brojOsoba = 0;
    float trosak, prosjecniDohodak, sumaDohodaka = 0.f;
    char jedIdRac[17+1];

    troskovi = fopen("troskovi.txt", "r");
    porezi = fopen("porezi.bin", "r+b");

    while (fread(&osoba, sizeof(osoba), 1, porezi) == 1) {
        ++rbrZapis;
        if (osoba.sifOsoba == rbrZapis) { /* osoba s tom sifrom postoji */
            sumaDohodaka += osoba.dohodak;
            ++brojOsoba;
        }
    }
    prosjecniDohodak = sumaDohodaka/brojOsoba;

    while (fscanf(troskovi, "%d%s%f", &sifOsoba, jedIdRac, &trosak) == 3) {
        fseek(porezi, (long)(sifOsoba - 1)*sizeof(osoba), SEEK_SET);
        fread(&osoba, sizeof(osoba), 1, porezi);
        if (osoba.dohodak < prosjecniDohodak) {
            osoba.porez = osoba.porez >= trosak ? osoba.porez - trosak : 0.f;
            fseek (porezi, -(long)sizeof(osoba), SEEK_CUR);
            fwrite(&osoba, sizeof(osoba), 1, porezi);
        }
    }
    fclose (troskovi);
    fclose (porezi);

    return 0;
}
```

4.

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

void umetniClan (float *niz, int n, float x){
    int i = 0, j;
    while (i < n && niz[i] >= x) {
        ++i;
    }
    if (i != n) {
        for (j = n - 1; j > i; --j) {
            niz[j] = niz[j - 1];
        }
        niz[i] = x;
    }
    return;
}

int main(void) {
    float polje[5] = {0.f};
    int i, j;
    float broj;
    srand((unsigned)time(NULL));
    for (i = 0; i < 10; ++i) {
        broj = (rand() % 1000) / 100.f;
        printf("Ubacujem %4.2f: ", broj);
        umetniClan(polje, 5, broj);
        for (j = 0; j < 5; ++j) {
            printf(" %4.2f", polje[j]);
        }
        printf("\n");
    }
    return 0;
}
```

5.

```
#include <stdio.h>

int main (void) {
    int datum = 0x497A;
    printf("Godina: %d\n", ((datum & 0xFE00) >> 9) + 1980);
    /* 0xFE00 je isto što i 0x0000FE00 */

    /* dobra su i sljedeća rješenja:
    printf("Godina: %d\n", ((datum >> 9) & 0x7F) + 1980);
    printf("Godina: %d\n", (datum >> 9) + 1980);
    printf("Godina: %d\n", (datum /512) + 1980);
    */

    return 0;
}
```

6.

```
#include <stdio.h>

int main (void) {
    struct {
        short int mbr;
        char prezime[20+1];
        int ocjene[2];
    } studenti[100];

    scanf("%hd%s%d%d", &studenti[1].mbr, studenti[1].prezime, &studenti[1].ocjene[0],
        &studenti[1].ocjene[1]);

    return 0;
}
```