

JMBAG		Prezime, ime	
-------	--	--------------	--

## Uvod u programiranje

### Završni ispit - 28. siječnja 2019.

Rješenja zadataka 1 - 4 napisati na vlastitim papirima i predati ih u košuljici.

Rješenja zadataka 5 - 7 napisati na za to predviđeno mjesto uz tekst zadatka te list sa zadacima predati u košuljici.

#### 1. (10 bodova)

Svaki zapis tekstne datoteke pred.txt sadrži šifru predmeta (cijeli broj) i naziv predmeta (niz znakova ne dulji od 30 znakova). Svaki zapis binarne datoteke pred.bin sadrži šifru predmeta (int), naziv predmeta (30+1 znak) i broj ECTS bodova (int). Redni broj zapisa u datoteci pred.bin odgovara šifri predmeta.

Napisati program kojim će se na zaslon ispisati predmeti iz pred.txt za koje postoji zapis s odgovarajućom šifrom predmeta u pred.bin, ali naziv predmeta u tom zapisu nije jednak nazivu predmeta u pred.txt.

Primjer sadržaja pred.txt

```
12·Fizika␣
5·Mehanika␣
2·BAZE·PODATAKA␣
10·Programiranje␣
11·Elektromehanika␣
6·Elektronika·1␣
3·Elektronika·2␣
7·Engleski·jezik␣
8·Matematika·II·dio␣
```

Primjer sadržaja pred.bin

2	Baze podataka	7
6	Elektronika 1	5
7	Engleski jezik	3
8	Matematika 2	6
10	Programiranje	6
11	Elektro-mehanika	5
12	Fizika	6

Primjer ispisa na zaslon

```
...2·BAZE·PODATAKA␣
...11·Elektromehanika␣
...8·Matematika·II·dio␣
```

#### 2. (10 bodova)

Napisati funkciju genSlovo tipa char. Funkcija nema niti jedan parametar. Uz korištenje generatora pseudoslučajnih brojeva, funkcija vraća slučajno odabrano veliko slovo kod svakog neparnog poziva, a slučajno odabrano malo slovo kod svakog parnog poziva. Npr. kod prvog poziva unutar jednog izvođenja programa, funkcija vraća slučajno odabrano slovo **N**, kod drugog poziva vraća slučajno odabrano slovo **u**, kod trećeg poziva vraća slučajno odabrano slovo **P**, itd. Generator pseudoslučajnih brojeva mora se inicijalizirati u pozivajućoj funkciji (npr. u funkciji main).

U nastavku, u istom modulu, napisati funkciju main ("glavni program") u kojoj će se uzastopnim pozivanjem funkcije genSlovo generirati niz znakova duljine točno 10 znakova (npr. niz znakova **NuPg0hXuUw**) te ga prema konverziskoj specifikaciji %s ispisati na zaslon.

#### 3. (10 bodova)

Napisati funkciju pokNaMax koja za zadanu dvodimenzijsku kvadratnu matricu reda n (članovi matrice su tipa double) vraća **pokazivač** na najveći član na glavnoj dijagonali matrice. Ako na glavnoj dijagonali postoji više članova koji imaju istu (najveću) vrijednost, funkcija vraća pokazivač na član koji ima manji redni broj retka.

Primjer: za matricu na slici, funkcija treba vratiti pokazivač na označeni član.

-1.	2.	3.	9.
-9.	5.	3.	-4.
7.	2.	2.	4.
1.	2.	-3.	5.

U glavnom programu inicijalizirati matricu na vrijednosti iz priloženog primjera (pomoću inicijalizatora, ne učitavanjem s tipkovnice) te pozivom funkcije pokNaMax odrediti i na zaslon ispisati **vrijednost** najvećeg člana na glavnoj dijagonali matrice.

**4. (9 bodova)**

Napisati funkciju `brojiZnamenke` koja vraća broj znamenki u zadanom znakovnom nizu.

**5. (2 boda)**

Prikazati što će se tijekom izvršavanja odsječka programa ispisati na zaslonu ako je preko tipkovnice upisan sadržaj prikazan u desnom okviru.

```
float x = 1.0f;
int n = 20;
char ime[20] = "Ivana";
scanf("%f%d", &x, &n);
printf("%5.1f\n", x);
printf("%s\n", ime);
printf("%d", n);
```

• 6.2-3Ana•Horvat•10↵

[illegible]

**6. (2 boda)**

Nadopuniti programski kôd (isključivo iznad crte) tako da rezultat funkcije fun bude jednak rezultatu funkcije strlen iz <string.h>.

```
int fun(char *s) {
    int n = 0;
    while ( _____ ) {
        ++n;
    }
    return n;
}
```

**7. (2 boda)**

U binarnoj datoteci naziva `dat.bin` ima ukupno 8 bajtova. Prikazati što će se tijekom izvršavanja odsječka programa ispisati na zaslonu.

```
short p[10];
FILE *tok = fopen("dat.bin", "rb");
printf("%d", fread(p, sizeof(p[0]), 10, tok));
```

[illegible]

## Rješenja:

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

int main(void) {
    struct pred_s {
        int sif;
        char naz[30 + 1];
        int ects;
    } pred;
    char naz[30 + 1];
    int sif;

    FILE *txt = fopen("pred.txt", "r");
    FILE *bin = fopen("pred.bin", "rb");

    while (fscanf(txt, "%d %[^\\n]", &sif, naz) == 2) {
        fseek(bin, (long)(sif - 1) * sizeof(pred), SEEK_SET);
        fread(&pred, sizeof(pred), 1, bin);
        if (pred.sif == sif && strcmp(pred.naz, naz) != 0) {
            printf("%4d %s\\n", sif, naz);
        }
    }
    fclose(txt);
    fclose(bin);
    return 0;
}
```

```
2. #include <stdio.h>
#include <stdlib.h>
#include <time.h>
#define BRSLOVA 10

char genSlovo(void) {
    static _Bool parniPoziv = 0;
    char c = rand() % ('Z' - 'A' + 1) + 'A';
    if (parniPoziv) {
        c = c + ('a' - 'A');
    }
    parniPoziv = !parniPoziv;
    return c;
}

int main(void) {
    srand((unsigned int)time(NULL));
    char niz[BRSLLOVA + 1];
    for (int i = 0; i < BRSLOVA; ++i) {
        niz[i] = genSlovo();
    }
    niz[BRSLOVA] = '\\0';
    printf("%s", niz);
    return 0;
}
```

}

}

1