

Uvod u programiranje**Ispit - grupa 3**

1. (**13p**) Napisati program koji učitava broj n, matricu dimenzija nxn i decimalan broj k. Na osnovu učitane matrice i broja k se pronađi i ispisuje odgovarajuća podmatrica. Potrebno je ispisati najveću podmatricu čija je aritmetička sredina (zbir svih elemenata podeljen sa brojem elemenata) jednak broju k. Ukoliko ne postoji takva podmatrica ispisati onu čija je aritmetička sredina elemenata najpričinjiva broju k. Nije dozvoljeno koristiti pomoćne nizove/matrice.

Primeri:

Ulaz: n=4 , k=4

```
4 4 1 2
4 4 3 5
7 8 9 6
2 3 4 4
```

Izlaz:

```
4 4
4 4
```

Ulaz: n=5 , k = 3.14

```
7 2 3 4 9
5 4 3 8 1
3 3 3 3 3
5 3 8 3 2
7 5 2 2 2
```

Izlaz:

```
3 3 3 3 3
```

Objašnjenje drugog primera:

Ne postoji podmatrica sa aritmetičkom sredinom 3.14. Najблиža sredina je 3.00 (razlika 0.14), a najveća takva podmatrica je čitav treći red (1x5):

2. (**12p**) Napisati program koji učitava jedan string sa rečima odvojenim razmacima i, bez korišćenja dodatnih stringova, menja ga na sledeći način:

- Reči dužine 5 karaktera: Izbaciti svaki njihov duplikat osim poslednjeg pojavljivanja.
- Reči dužine 4 karaktera: Zameniti ih prvim narednim rečima dužine 6 karaktera, biranim ciklično (ako posle reči od 4 karaktera nema nijedne reči od 6 karaktera, nastaviti pretragu od početka stringa).

Na kraju, izmenjeni string odštampati na standardni izlaz.

Primer:

Ulaz:

apple apple mango code python code hello

Izlaz:

apple mango python python python hello

Objašnjenje:

Reč apple se pojavljuje dva puta, pa zadržavamo samo njeno poslednje pojavljivanje.

Prva reč od 4 karaktera je code, pa je zamenjujemo prvom narednom rečju od 6 karaktera (to je python). Druga reč code ne nalazi više nijednu reč od 6 karaktera posle sebe, pa se ciklično vraćamo na početak i nalazimo python kod prvog pojavljivanja.

3. (25p) Deklaracija strukture je data ovako (ne menjati!):

```
typedef struct Pokemon {
    char ime[50]; int cp; char tip[20]; struct Pokemon *sledeci;
} Pokemon;
```

U tekstualnoj datoteci "pokemon.txt" nalaze se podaci o Pokemonima. Za svakog Pokemona su zabeležene tačno tri uzastopne linije:

Ime Pokemona (string bez razmaka, npr. Pikachu)

Combat power (CP) (celobrojna vrednost)

Tip Pokemona (string, npr. Electric, Water, Fire)

U datoteci zadatka nalaze se prazne implementacije sledećih funkcija. Ne menjati potpis nijedne od njih, niti glavni program.

- a) (10p) **void ucitajPokemone(Pokemon **lista)** - Učitava sve zapise iz "pokemon.txt" u jednostruko povezanu listu i vraća pokazivač na njen početak. Lista mora biti sortirana opadajuće po vrednosti cp (Pokemon sa najvišim CP-om na početku).
- b) (2p) **void ispisiListu(Paket *lista)** - Ispisuje sve Pokemone redom od početka ka kraju, u istom formatu kao u fajlu (ime, novi red; cp, novi red; tip, novi red).
- c) (4p) **int zbirCPpoTipu(Pokemon *lista, char *tip)** - Vraća zbir svih cp vrednosti Pokemona čiji je tip jednak prosleđenom tipu. Ako nema nijednog takvog Pokemona, vraća 0.
- d) (7p) **int izbaciParam(Pokemon **lista, char *tip)** - Izbacuje iz liste (i pravilno oslobađa memoriju) sve Pokemone sa datim tipom. Vraća broj izbačenih Pokemona.
- e) (2p) **Pokemon* obrisiListu(Pokemon *lista)** - Briše celu listu i oslobađa svu zauzetu memoriju.

Napomena: Dozvoljeno je dodavati pomoćne funkcije unutar .c fajla.

Primer ispisa za treći zadatak:

Početak ispisa:

Charizard

1500

... (ostatak pokemona)

Eevee

400

Normal

Kraj prvog ispisa i prelaz na operacije:

Ukupan CP za tip Electric: 1700

Ukupan CP za tip Water: 1900

Posle izbacivanja Pokemona tipa Water:

Ukupan CP za tip Water: 0

(nastavak ispisa na desnom delu papira...)

(...nastavak ispisa)

Charizard

1500

Fire

... (ostatak pokemona)

Eevee

400

Normal

Kraj drugog ispisa i brisanje:

Posle brisanja

(nil)

Spisak dozvoljenih bibliotečkih funkcija (u nastavku) važi generalno. Dozvoljeno je implementirati svoju verziju bibliotečkih funkcija koje nisu dozvoljene.

- **stdio.h:** printf, scanf, fscanf, fgets, puts, fputs
- **string.h:** strlen, strcpy, strcat, strstr, strchr, strcmp, strtok
- **math.h:** sqrt, abs, pow, ceil, floor, round
- **slib.h:** malloc, calloc, realloc, free, atoi, atof