

Uvod u programiranje

Ispit – julski rok – grupa 2

1. (13p) Sa konzole se unosi broj N i matrica M dimenzija $N \times N$. Potrebno je pronaći najduže nizove prilikom spiralnog obilaska matrice koji su:

- 1) monotono rastući: svaki sledeći element je sve veći
- 2) monotono opadajući: svaki sledeći element je sve manji
- 3) stagnirajući: svi elementi su jednaki

Potrebno je pronaći najduži niz za sve tri od navedenih kategorija. Ukoliko ima više nizova koji su najduži:

- 1) u kategoriji monotono rastućih i monotono opadajućih nizova pronaći one kod kojih je apsolutna razlika između prvog i poslednjeg elementa najveća; ako je i ta najveća razlika jednaka kod različitih nizova, izvući nizove koji su prvi pronađeni prilikom obilaska
- 2) u kategoriji stagnirajućih nizova izvući onaj koji je prvi pronađen prilikom obilaska

Za pronađeni niz ispisati njegove elemente.

Napomena: Nije dozvoljeno korišćenje pomoćnih nizova i matrica.

Ulaz:

```
N = 3
2 6 9
8 6 9
8 2 1
```

Izlaz:

```
monoton rastuci 2 6 9
monoton opadajuci 9 1
stagnirajuci 9 9
```

2. (12p) Napisati program koji iz konzole učitava string sastavljen od malih slova, razmaka, znakova tačka i znakova zarez. Potrebno je modifikovati string na sledeći način i ispisati ga nakon modifikacija:

- 1) ako se na kraju neke reči nalaze znaci zarez, potrebno je da se ta reč ponovi onoliko puta koliko ima tog znaka na kraju reči, pri čemu se znaci zarez brišu iz stringa
- 2) ako se na kraju neke reči nađu znaci tačka, potrebno je ceo podstring između prethodnih i tih tačaka ponavljaju onoliko puta koliko ima znakova tačke, pri čemu se znaci tačka razdvajaju tako da se na kraju svakog podstringa pojave tačno jednom

Primer:

Ulaz: ja sam ucio,, i valjda cu poloziti... ali ne znam.

Izlaz: ja sam ucio ucio i valjda cu poloziti. ja sam ucio ucio i valjda cu poloziti. ja sam ucio ucio i valjda cu poloziti. ali ne znam.

Nije dozvoljeno korišćenje dodatnih nizova (uključujući stringove), kao i korišćenje strtok funkcije u ovom zadatku.

3. (25p) Deklarisane je struktura Element na sledeći način:

```
typedef struct Element {
    int grupa; char modul[50]; int brojStudenata; char predmeti[5][50];
    struct Element *sledeci;
} Element;
```

U tekstuallnoj datoteci "ulaz.txt" nalazi se lista izbornih grupa na studijama smerova RN i RI. U prvoj liniji datoteke zadati su podaci razdvojeni tačka-zarezom: broj (oznaka) grupe, naziv modula i broj studenata. U narednih pet linija su predmeti koje ta grupa obuhvata. Svaka naredna grupa je zadata u istom formatu sa po šest linija u datoteci. Moguće je da grupa nema modul, u tom slučaju se u tom redu za naziv modula pojavljuje praznina.

U fajlu zadatka (koji se kompajlira ali ne radi ništa) je dat šablon za funkcije koje treba implementirati (ne menjati im potpis!) i glavni program koji ne treba menjati. Implementirati sledeće funkcije (dozvoljeno je dodavati i pomoćne funkcije):

a) (10) **Element * ucitajFajl()** - učitava datoteku u listu i vraća pokazivač na početak liste. Podaci o grupama se mogu učitati u proizvoljnem redosledu, nije ih potrebno sortirati. Voditi računa o praznom redu na kraju datoteke. Ako je lista prazna, njen početak je NULL.

b) (2) **void ispisiPodataka(Element *lista, char *smer, int godina)** - ispisi podataka o grupama po smeru i godini studija. Argumenti funkcije su kratak string (RN ili RI) koji predstavlja smer, i broj (3 ili 4) koji predstavlja godinu studija. U oznakama grupe, godina je cifra stotina a smer je cifra desetica (RN = 0, RI = 2), npr. grupa 321 je treća godina smer RI. Potrebno je ispisati sve podatke o svim grupama za uneti smer i godinu studija. Ako smer ili godina nisu ispravni, ispisati poruku o grešci.

c) (3) **void predefinisaneGrupe(Element *lista)** - za svaki smer ispisuje koje su predefinisane grupe. Grupa je predefinisana ako odgovarajući modul sa istim nazivom postoji i na trećoj i na četvrtoj godini na istom smeru. Grupa koja nema modul ne može biti predefinisana grupa. Dovoljno je ispisati samo broj-oznaku svake predefinisane grupe.

d) (8) **void obavezniPredmeti(Element *lista)** - za svaki smer i godinu studija potrebno je ispisati obavezne predmete na tom smeru i toj godini. Predmet je obavezan na smeru i godini studija ako ga slušaju sve grupe na tom smeru i godini studija.

e) (2) **void brisanje(Element **lista)** - briše sve elemente liste (oslobađa svu zauzetu memoriju), postavlja početak liste na NULL.

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

- **stdio.h:** printf, scanf, fgets, puts, fputs
- **stdlib.h:** malloc, calloc, realloc, free, atoi, atof
- **string.h:** strlen, strcpy, strcat, strstr, strchr, strcmp, strtok (strtok ne u drugom zadatku)
- **math.h:** sqrt, abs, pow, ceil, floor, round

Primer ispisa za treći zadatak (obratite pažnju da redosled ispisa zavisi od redosleda učitavanja)

Početak ispisa:

RN 3. godina:

Grupa: 306, Modul: , Broj studenata: 21

Verovatnoca i statistika

Programski prevodioci

Softverske komponente

Paralelni algoritmi

Skript jezici

Grupa: 305, Modul: QA, Broj studenata: 24

Verovatnoca i statistika

Programski prevodioci

Softverske komponente

Testiranje softvera

Paralelni algoritmi

Grupa: 304, Modul: Razvoj softvera, Broj studenata: 42

Verovatnoca i statistika (...)

Kraj drugog ispisa i prelaz na druge operacije:

Nevalidan smer:

Nevalidan smer

Nevalidna godina:

Nevalidna godina

Predefinisane grupe:

423

422

421

404

403

402

401

323

322

321

304

303

302

301

Obavezni predmeti:

RN 3. godina:

Verovatnoca i statistika

Programski prevodioci

RN 4. godina:

Upravljanje projektima

Kombinatorika i teorija grafova

RI 3. godina:

Verovatnoca i statistika
Primenjena vestacka inteligencija
RI 4. godina:
Bezicne i mobilne komunikacije
Upravljanje projektima

Brisanje...

Posle brisanja
0000000000000000