

## PROGRAMIRANJE 2 — RANIJI 2. KOLOKVIJ

24. lipnja 2009.

**Upute:** Na kolokviju je dozvoljeno koristiti samo pribor za pisanje i brisanje, te službeni šalabahter. Kalkulatori, mobiteli, razne neslužbene tablice, papiri i sl., **nisu** dozvoljeni!

Sva rješenja napišite isključivo na papire sa zadacima, jer jedino njih predajete. Ne zaboravite se **potpisati** na svim papirima! Skice smijete raditi i na drugim papirima koje će vam dati dežurni asistent.

U svim zadacima zabranjeno je korištenje dodatnih nizova i standardne matematičke biblioteke (zaglavlje `math.h`), osim ako je u zadatku drugačije navedeno.

Rezultati i uvid u kolokvije: **utorak, 30. lipnja 2009. u 12 sati.**

## ZADATAK 1

1

(10 bodova).

Napišite funkciju `strnstr` koja prima tri argumenta. Prva dva argumenta su konstantni stringovi `s` i `t`, a zadnji argument je cijeli broj `n`, za kojeg pretpostavljamo da je nenegativan (ne treba provjeravati).

Funkcija treba provjeriti pojavljuje li se cijeli string `t` kao podstring u stringu `s`, s tim da **početak** tog podstringa smije biti samo među prvih `n` znakova u `s`. Dozvoljeno je da string `s` ima i manje od `n` znakova. Ako takav podstring postoji, funkcija treba vratiti pokazivač na prvi znak tog podstringa u `s`, a u protivnom vraća `NULL`.

Zabranjeno je korištenje funkcija iz biblioteke `string.h`!

## PROGRAMIRANJE 2 — RANIJI 2. KOLOKVIJ — ZADATAK 2

24. lipnja 2009.

(10 + 5 = 15 bodova).

Riječ u nekom tekstu je svaki niz uzastopnih znakova koji počinje otvorenom zagradom (, iza toga dolaze samo velika slova, s tim da barem jedno slovo mora biti prisutno, a završava zatvorenom zagradom ).

- (a) Napišite funkciju `najdulja` koja kao argument prima pokazivač na već otvorenu ulaznu tekst-datoteku. Funkcija treba u toj datoteci pronaći najdulju riječ koja zadovoljava zadana pravila, a zatim kreirati string u kojeg treba iskopirati tu riječ, i vratiti pokazivač na taj string. Ako takve riječi nema, funkcija vraća prazan string (ima samo nul-znak). Za izlazni string alocirajte točno onoliko memorije koliko je najmanje potrebno. Uputa: u prvom prolazu pamтите broj pročitanih znakova do početka riječi. Funkcija za pozicioniranje na početak datoteke je `void rewind(FILE *fp);`.
- (b) Napišite i program koji s komandne linije učitava ime datoteke, a zatim otvara za čitanje datoteku s tim imenom. Ako je datoteka uspješno otvorena, treba pozvati funkciju `najdulja` i na standardni izlaz ispisati nađenu najdulju riječ. Pazite da program oslobodi svu zauzetu memoriju!

## PROGRAMIRANJE 2 — RANIJI 2. KOLOKVIJ — ZADATAK 3

24. lipnja 2009.

(4 + 6 + 15 = 25 bodova).

Za obradu rezultata kolokvija, podatke o jednom studentu pamtimo u obliku strukture koja sadrži 4 člana: JMBAG (polje od 10 znakova), ime i prezime studenta (polje od 40 znakova), bodove po zadacima (polje od 5 cijelih brojeva) i ukupne bodove (jedan cijeli broj).

- (a) Definirajte tip `student` za takvu strukturu, koja sadrži samo opisane članove, tako da bude moguća deklaracija `student s;`.

Napišite sljedeće dvije funkcije i primjere poziva za svaku od njih.

- (b) Funkcija `zbroji` kao argument dobiva jednu strukturu tipa `student`. Pretpostavljamo da ta struktura sadrži sve podatke, osim ukupnog broja bodova. Funkcija treba zbrojiti bodove po svim zadacima i vratiti tu strukturu s korektno postavljenim ukupnim brojem bodova.
- (c) Funkcija `prošli` kao argumente prima dva stringa, koji predstavljaju ime ulazne i ime izlazne datoteke studenata. Pretpostavljamo da ulazna datoteka sadrži podatke o studentima **bez** ukupnog broja bodova. Funkcija treba svim studentima u ulaznoj datoteci postaviti ukupni broj bodova, koristeći direktno binarno čitanje i pisanje, te funkciju `zbroji`. Dodatno, u novu izlaznu datoteku treba prepisati sve one studente koji ukupno imaju barem 45 bodova. Izlazna vrijednost funkcije je broj studenata u novoj datoteci. Ako prilikom otvaranja bilo koje datoteke dođe do greške, funkcija treba vratiti `-1`. Ostale moguće greške ignorirajte.

Uputa: funkcija za pozicioniranje u datoteci je

```
int fseek(FILE *fp, long offset, int origin);
```

gdje je `offset` pomak u byteovima, a `origin` jedna od vrijednosti `SEEK_SET`, `SEEK_CUR` ili `SEEK_END`. Izlaz različit od nule je signal greške.

Olakšanje (za 5 bodova manje): Ako ne znate napraviti postavljanje ukupnog broja bodova pozicioniranjem u ulaznoj datoteci, pretpostavite da ulazna datoteka već **sadrži** i ukupne bodove za svakog studenta, pa samo kreirajte novu datoteku po zadanom pravilu.

## PROGRAMIRANJE 2 — RANIJI 2. KOLOKVIJ — ZADATAK 4

24. lipnja 2009.

(4 + 6 + 10 + 10 = 30 bodova).

Kružnica u ravnini  $\mathbb{R}^2$  zadana je koordinatama  $x_1, y_1$  i  $x_2, y_2$  točaka  $T_1$  i  $T_2$  na suprotnim krajevima dijametra te kružnice.

- (a) Definirajte tip **kružnica** koji prikazuje jedan čvor (element) vezane liste takvih kružnica. U strukturi smiju biti samo članovi koji su nužni za opisani prikaz kružnice i vezane liste.

Napišite sljedeće funkcije i primjere poziva za svaku od njih.

- (b) Funkcija **desno** kao argument prima jedan čvor tipa **kružnica**. Funkcija treba vratiti vrijednost 1 ako se **cijela** zadana kružnica nalazi u desnoj otvorenoj poluravnini. U suprotnom, funkcija treba vratiti 0. Uputa: izračunajte središte i polumjer kružnice. Ovdje smijete koristiti funkcije iz matematičke biblioteke (zaglavlje **math.h**).
- (c) Funkcija **LijeveSlike** kao argument prima pokazivač na vezanu listu kružnica. Funkcija treba kreirati novu vezanu listu kružnica na sljedeći način: za svaku kružnicu iz prve liste koja je sadržana u desnoj otvorenoj poluravnini, treba kreirati novi čvor koji sadrži kružnicu koja je simetrična polaznoj obzirom na  $y$ -os. Funkcija treba vratiti pokazivač na prvi element nove liste. Poredak kružnica u novoj listi nije bitan.
- (d) Funkcija **rastavi** kao argument prima pokazivač na vezanu listu kružnica. Tu listu treba rastaviti na dvije liste (samo promjenama pokazivača, bez kreiranja novih elemenata), tako da u prvoj listi ostanu samo kružnice sadržane u desnoj otvorenoj poluravnini, a u drugoj se nalaze sve ostale. Pokazivač na prvu listu je izlazna vrijednost funkcije, a pokazivač na drugu treba vratiti kao varijabilni argument.

U (c) i (d) dijelu zadatka smijete koristiti funkciju **desno**, čak i ako ju niste napisali.