

## PROGRAMIRANJE II (22.06.2022)

❶ (20 bodova) Neka je dat tip:

```
typedef struct {  
    double x, y;  
} POINT;
```

kojim se reprezentuje tačka u dvodimenzionom koordinatnom sistemu.

Napisati funkciju *open* sa promjenljivim brojem argumenata koja prihvata string *m*, cijeli broj *n* i *n* stringova (neobavezni argumenti) koji predstavljaju nazive datoteka, a vraća dinamički niz od *n* otvorenih datoteka čiji su nazivi neobavezni argumenti. Parametar *m* predstavlja mod rada u kojem datoteke treba da budu otvorene.

Napisati funkciju *write* koja prihvata niz od *n* otvorenih ulaznih binarnih datoteka u kojima je upisan nepoznat broj podataka o tačkama (podaci tipa POINT). Funkcija treba da u tekstualnu datoteku, čiji je naziv treći parametar funkcije, upiše podatke o tačkama iz svih ulaznih datoteka. Podaci o svakoj tački treba da budu upisani u zasebnom redu u formatu:

(x,y)

gde su *x* i *y* koordinate tačke. Funkcija treba da vrati ukupan broj upisanih podataka o tačkama.

❷ (20 bodova) Neka je dat tip STUDENT i funkcija *cmpStudent* koja poredi dva podatka tipa STUDENT. Funkcija *cmpStudent* prihvata adrese dva podatka tipa STUDENT, a vraća vrijednost -1 (prvi podatak manji od drugog), 0 (podaci su jednaki) ili 1 (prvi podatak veći od drugog). Prototip funkcije je:

```
int cmpStudent(STUDENT *podatak1,  
               STUDENT *podatak2);
```

Napisati funkciju (korištenjem funkcije *cmpStudent*) koja vrši sortiranje niza od *n* podataka tipa STUDENT. Niz treba da se sortira u opadajućem redoslijedu prema vrijednostima elemenata originalnog niza. Sortiranje treba izvršiti *bubble-sort* algoritmom. Prototip funkcije je:

```
void sortStudent(STUDENT *niz, int n);
```

Napisati funkciju koja vrši binarnu rekurzivnu pretragu za dati niz studenata koristeći funkciju *cmpStudent* i vraća indeks pronađenog člana niza, a čiji je prototip:

```
int binSearchRek(STUDENT *niz, STUDENT  
key, int (cmpStudent*)(STUDENT  
*podatak1, STUDENT *podatak2), int begin,  
int end);
```

pri čemu je parametar *key* student koji se traži. Smatrati da u datom nizu nema duplikata po datoj funkciji poređenja.

❸ (20 bodova) Neka je dat tip DOCUMENT.

Napisati program u kojem treba:

1. Definirati tip DOCUMENT\_NODE kojim se reprezentuje čvor odgovarajuće strukture podataka koja se koristi za čuvanje dokumenata. Odabrati strukturu tako da memorijsko zauzeće bude što efikasnije, odnosno da se omogući čuvanje što većeg broja dokumenata, smatrajući da će se oni rijetko pretraživati.
2. Implementirati funkciju za dodavanje čvora na kraj strukture.
3. Implementirati funkciju za dodavanje čvora na početak strukture.
4. Implementirati funkciju za brisanje čvora iza zadatog čvora.

**Napomena:** Na početku zadatka napisati koja je struktura odabrana.

❹ (20 bodova) Definirati tip NODE kojim se reprezentuje čvor stabla binarnog pretraživanja, čiji je informacioni sadržaj cijeli broj.

Napisati funkciju koja na osnovu zadatog niza (parametar *array*) formira i vraća stablo binarne pretrage. Prototip funkcije je:

```
NODE *form(int *array, int n);
```

Napisati rekurzivnu funkciju koja obilazi stablo u *inorder* poretku i ispisuje samo one čvorove za koje je ispunjen uslov. Prototip funkcije je:

```
void inorder(NODE *root, int  
(*condition)(NODE *n));
```

gdje je *condition* funkcija koja vrši provjeru informacionog sadržaja pri posjeti čvora i vraća 1 ako je uslov ispunjen, a 0 ako nije za zadati čvor.

Napisati glavni program u kojem će se ilustrovati kreiranje stabla sa sljedećim čvorovima 3, 17, 5, 9 (koristiti funkciju *form*) i ispisati u *inorder* poretku samo one čvorove čiji je informacioni sadržaj veći od 6 (koristiti funkciju *inorder*). Prikazati (nacrtati) izgled stabla nakon svakog umetanja.