

**PROGRAMIRANJE II - K1
(12.04.2023)**

- ❶ (25 bodova) Napisati funkciju koja računa broj načina na koji se možemo popeti uz n stepenica. U jednom koraku možemo preći 1, 2 ili 3 stepenice. Ukoliko smo u trenutnom koraku prešli 2 ili 3 stepenice, u sljedećem koraku možemo preći samo 1 stepenicu. Prototip funkcije je:

int countClimbWays(int n, int lastStep);

- ❷ (25 bodova) Napisati funkciju *write* sa promjenljivim brojem argumenata, koja prihvata pokazivač na funkciju *wt*, cijeli broj n te n uređenih parova neobaveznih argumenta (dakle, ukupno $2n$ neobaveznih argumenata). Svaki uređeni par neobaveznih argumenata čini adresu nekog podatka (*void **) i naziv datoteke (*string*). Funkcija na koju pokazuje *wt* omogućava upis podatka čija je adresa prvi parametar te funkcije, u tekstualnu datoteku koja je drugi parametar te funkcije. Funkcija na koju pokazuje *wt* vraća informaciju o tome da li je upis podatka uspješno izvršen. Funkcija *write* treba da za svaki par neobaveznih argumenata, korištenjem funkcije na koju pokazuje *wt*, izvrši upis podatka čija je adresa prvi član para, u tekstualnu datoteku čiji je naziv drugi član para. Funkcija *write* treba da vrati ukupan broj uspješno upisanih podataka. Prototip funkcije je:

int write(int (*wt)(void *, FILE *), int n, ...);

- ❸ (25 bodova) Neka je dat tip:

```
typedef struct predmet {  
    char id[15]; //identifikator  
    char naziv[64];  
    int ects;  
} PREMETS;
```

kojim se reprezentuju informacije o jednom predmetu.

Napisati funkciju koja prihvata nesortiran niz od n podataka o predmetima te naziv izlazne tekstualne datoteke. Funkcija treba da u izlaznu tekstualnu datoteku upiše podatke o predmetima sortirane u rastućem redoslijedu prema nazivu (primjenom *insert-sort* algoritma), pri čemu originalni niz podataka ne smije biti promijenjen. Prototip funkcije je:

**void sortiraj(PREMET *niz, int n,
const char *dat);**

- ❹ (25 bodova) Neka je dat tip:

```
typedef struct student {  
    char broj_indeksa[6];  
    char ime[64];  
    char prezime[64];  
} STUDENT;
```

kojim se reprezentuje informacija o jednom studentu. Studenti su upisani u određenu binarnu datoteku tako da su sortirani po broju indeksa. Napisati funkciju čiji je prototip:

void obrisiStudenta(FILE *file, char indeks[]);

Funkcija briše iz binarne datoteke (parametar *file*) studenta sa brojem indeksa (parametar *indeks*). Prilikom pretrage studenta potrebno je vršiti binarnu pretragu studenta po broju indeksa. Smatrati da je datoteka otvorena. U datoteci na kraju mora biti upisano novo stanje tj. lista studenata bez traženog studenta.

**PROGRAMIRANJE II - K1
(12.04.2023)**

- ❶ (25 bodova) Napisati funkciju koja računa broj načina na koji se možemo popeti uz n stepenica. U jednom koraku možemo preći 1, 2 ili 3 stepenice. Ukoliko smo u trenutnom koraku prešli 2 ili 3 stepenice, u sljedećem koraku možemo preći samo 1 stepenicu. Prototip funkcije je:

int countClimbWays(int n, int lastStep);

- ❷ (25 bodova) Napisati funkciju *write* sa promjenljivim brojem argumenata, koja prihvata pokazivač na funkciju *wt*, cijeli broj n te n uređenih parova neobaveznih argumenta (dakle, ukupno $2n$ neobaveznih argumenata). Svaki uređeni par neobaveznih argumenata čini adresu nekog podatka (*void **) i naziv datoteke (*string*). Funkcija na koju pokazuje *wt* omogućava upis podatka čija je adresa prvi parametar te funkcije, u tekstualnu datoteku koja je drugi parametar te funkcije. Funkcija na koju pokazuje *wt* vraća informaciju o tome da li je upis podatka uspješno izvršen. Funkcija *write* treba da za svaki par neobaveznih argumenata, korištenjem funkcije na koju pokazuje *wt*, izvrši upis podatka čija je adresa prvi član para, u tekstualnu datoteku čiji je naziv drugi član para. Funkcija *write* treba da vrati ukupan broj uspješno upisanih podataka. Prototip funkcije je:

int write(int (*wt)(void *, FILE *), int n, ...);

- ❸ (25 bodova) Neka je dat tip:

```
typedef struct predmet {  
    char id[15]; //identifikator  
    char naziv[64];  
    int ects;  
} PREMETS;
```

kojim se reprezentuju informacije o jednom predmetu.

Napisati funkciju koja prihvata nesortiran niz od n podataka o predmetima te naziv izlazne tekstualne datoteke. Funkcija treba da u izlaznu tekstualnu datoteku upiše podatke o predmetima sortirane u rastućem redoslijedu prema nazivu (primjenom *insert-sort* algoritma), pri čemu originalni niz podataka ne smije biti promijenjen. Prototip funkcije je:

**void sortiraj(PREMET *niz, int n,
const char *dat);**

- ❹ (25 bodova) Neka je dat tip:

```
typedef struct student {  
    char broj_indeksa[6];  
    char ime[64];  
    char prezime[64];  
} STUDENT;
```

kojim se reprezentuje informacija o jednom studentu. Studenti su upisani u određenu binarnu datoteku tako da su sortirani po broju indeksa. Napisati funkciju čiji je prototip:

void obrisiStudenta(FILE *file, char indeks[]);

Funkcija briše iz binarne datoteke (parametar *file*) studenta sa brojem indeksa (parametar *indeks*). Prilikom pretrage studenta potrebno je vršiti binarnu pretragu studenta po broju indeksa. Smatrati da je datoteka otvorena. U datoteci na kraju mora biti upisano novo stanje tj. lista studenata bez traženog studenta.