

PROGRAMIRANJE II – K1–GRUPA B
(13.04.2022)

- ❶ (25 bodova) Neka je data matrica dimenzija $m \times n$. Napisati rekurzivnu funkciju koja računa ukupan broj putanja kojima se može doći od posljednje $(m-1, n-1)$ do prve ćelije u matrici $(0, 0)$. Moguće putanje za naredni korak su lijevo i gore u odnosu na početnu ćeliju. Smatrati da iz prvog reda (0) ili prve kolone (0) postoji samo jedna putanja do prve ćelije. Prototip funkcije je:

```
int izracunajPutanje(int currentRow, int  
currentColumn, int m, int n)
```

- ❷ (25 bodova) Neka je dat tip:

```
typedef struct artikal {  
    char sifra[15];  
    char naziv[64];  
    double kol, cijena;  
} ARTIKAL;
```

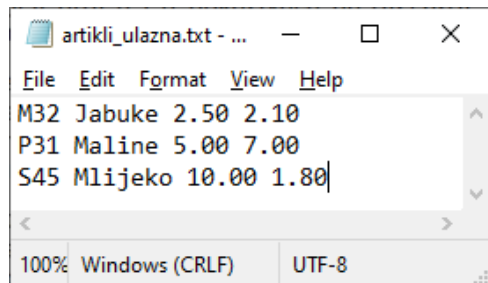
kojim se reprezentuju informacije o jednom artiklu.

Napisati funkciju *provjeri_datoteku* koja za zadati artikal (parametar *a*) provjerava da li se nalazi u tekstualnoj datoteci sa nazivom "artikli_ulazna.txt". Ukoliko su podaci o artiklu upisani u datoteku, funkcija treba da vrati 1, inače treba da vrati 0. Artikli su jednaki ukoliko su im jednake šifre. U svakom redu datoteke je upisan podatak o jednom artiklu u sljedećem formatu: *sifra naziv kol cijena*.

Prototip funkcije *provjeri_datoteku* je:

```
int provjeri_datoteku(ARTIKAL *a);
```

Primjer ulazne datoteke je prikazan na sljedećoj slici:



Napisati funkciju *func* sa promjenljivim brojem argumenata koja prihvata pokazivač na funkciju (parametar *pf*), cijeli broj *n* i *n* pokazivača na ARTIKAL (neobavezni argumenti).

Parametar *pf* predstavlja pokazivač na funkciju koja provjerava da li je ispunjen neki uslov za dati ARTIKAL (funkcija vraća vrijednost 0 ako uslov nije ispunjen, a vrijednost 1 ako uslov jeste ispunjen).

Funkcija *func* treba da podatke o artiklima (neobavezni argumenti) koji ispunjavaju dati uslov upiše u binarnu datoteku sa nazivom "artikli_izlazna.dat".

Prototip funkcije *func* je:

```
void func(int (*pf)(ARTIKAL *), int n,...);
```

- ❸ (25 bodova) Neka je dat tip SLIKA i funkcija *cmpimg* koja poređi dva podatka tipa SLIKA. Funkcija *cmpimg* prihvata adrese dva podatka tipa SLIKA, a vraća vrijednost -1 (prvi podatak manji od drugog), 0 (podaci su jednaki) ili 1 (prvi podatak veći od drugog). Prototip funkcije je:

```
int cmpimg(const SLIKA *podatak1, const SLIKA  
*podatak2);
```

Napisati funkciju (korištenjem funkcije *cmpimg*) koja vrši indirektno (originalni niz ne smije biti modifikovan) sortiranje niza od *n* podataka tipa SLIKA. Funkcija treba da vrati dinamički niz adresa elemenata originalnog niza sortiranih u rastućem redoslijedu prema vrijednostima elemenata originalnog niza. Sortiranje treba izvršiti *selection-sort* algoritmom. Prototip funkcije je:

```
SLIKA** sortimg(SLIKA *niz, int n);
```

- ❹ (25 bodova) Neka je dat tip:

```
typedef struct osoba {  
    char ime[64];  
    char prezime[64];  
    double visina;  
} OSOBA;
```

Napisati funkciju koja vrši binarnu pretragu osoba po svim poljima tipa OSOBA, a čiji je prototip:

```
OSOBA* pretraga(OSOBA *niz, int  
*sortiraniIndeksi, int n)
```

a koja vraća adresu osobe ukoliko je pronađena, u suprotnom se vraća vrijednost NULL. Pri tome parametar *niz* predstavlja adresu niza osoba (osobe u nizu nisu sortirane), parametar *sortiraniIndeksi* predstavlja adresu niza u kojem su sortirani indeksi originalnog niza (indirektno sortiranje po pravilu prvo ime, pa prezime pa visina osobe iz originalnog niza) i parametar *n* koji predstavlja dužinu prethodno navedenih nizova.