

PROGRAMIRANJE II (13.10.2020.)

- ❶ (15 bodova) Potrebno je definisati funkciju **filter_words** u kojoj se riječi, koje zadovoljavaju određeni uslov, kopiraju iz jedne tekstualne datoteke u drugu tekstualnu datoteku. Svaka kopirana riječ se upisuje u novi red izlazne datoteke. Prototip funkcije je sljedeći:

```
void filter_words(int (*uslov)(char *),  
FILE *in, FILE *out);
```

Funkcija kao argumente prima: funkciju koja se koristi za ispitivanje ispunjenosti uslova za određenu riječ (povratna vrijednost iz funkcije je 1 ukoliko je uslov ispunjen, u protivnom je 0), ulaznu i izlaznu datoteku.

U glavnoj funkciji (**main**) potrebno je demonstrirati upotrebu **filter_words** funkcije kopiranjem samo onih riječi koje sadrže maksimalno pet znakova. Nazivi ulazne i izlazne datoteke prosljeđuju se kao argumenti komandne linije.

- ❷ (15 bodova) Neka je dat sljedeći tip podataka:

```
typedef struct {  
    int serial_number;  
    char name[31];  
    int in_stock;  
} PRODUCT;
```

Potrebno je:

- iz datoteke, čiji se naziv prosljeđuje putem standardnog ulaza, pročitati binarno upisane podatke o nepoznatom broju proizvoda i kreirati dinamički niz,
- primjenom *selection-sort* algoritma sortirati proizvode po serijskom broju u rastućem redoslijedu,
- upotrebom binarnog pretraživanja pronaći proizvod čiji serijski broj odgovara onom koji se unosi putem standardnog ulaza i
- pronađeni proizvod formatirano ispisati na standardni izlaz.

Napomena: ukoliko se ne pronađe traženi proizvod, potrebno je na standardni izlaz ispisati odgovarajuću poruku.

Primjer formatiranog ispisa na standardni izlaz za pronađeni proizvod sa serijskim brojem 31313:

```
=====
SN      NAME                               INS
=====
31313   Lenovo_ThinkPad_T15_Gen_1         1
=====
```

- ❸ (15 bodova) Potrebno je:

- definisati tip CVOR kojim se reprezentuje čvor jednostruko povezane liste, pri čemu je informacioni sadržaj cjelobrojnog tipa;
- pretpostaviti da postoji implementirana funkcija **dodaj** kojom se elementi niza, koji se prosljeđuje kao argument funkcije, dodaju na kraj liste. Prototip funkcije je:

```
void dodaj(CVOR **pl, int *niz, int niz_size);
```
- definisati funkciju **brisi** kojom se brišu svi elementi iz liste koji nisu sadržani u nizu, koji se prosljeđuje kao argument funkcije. Prototip funkcije je:

```
void brisi(CVOR **pl, int *niz, int niz_size);
```
- napisati glavnu funkciju (**main**) u kojoj se demonstrira rad funkcije **brisi**.

- ❹ (15 bodova) Korištenjem netežinskog, usmjerenog grafa vizuelno se prikazuju autobuska stajališta gradskog prevoza. Čvor grafa predstavlja autobusko stajalište, pri čemu je informacioni sadržaj čvora oznaka stajališta (cijeli broj). Grana koja spaja dva čvora/stajališta označava da postoji relacija/veza između njih.

Potrebno je:

- koristiti matričnu reprezentaciju grafa;
- putem standardnog ulaza učitati:
 - dimenzije matrice susjednosti,
 - matricu susjednosti grafa,
 - oznaku početnog stajališta i
 - oznaku krajnjeg stajališta;
- korištenjem algoritma za obilazak grafa po dubini (DFS algoritam) provjeriti da li je krajnje stajalište dostižno početnom stajalištu i
- ispisati rezultat na standardni izlaz u glavnoj (**main**) funkciji.