

PROGRAMIRANJE II (28.01.2020.)

- ❶ (10 bodova) Napisati funkciju koja ima sljedeći prototip:

```
int saberi(int n, int (*f)(int, int), ...);
```

Funkcija kao argumente prima broj cjelobrojnih vrijednosti n , pokazivač na funkciju koja vrši sabiranje dvije cjelobrojne vrijednosti i vraća rezultat, te karaktere kao neobavezne argumente, respektivno. Funkcija kao rezultat vraća sumu brojeva koji su nastali konverzijom u cjelobrojnu vrijednost karaktera kao neobaveznih argumenata (ako se karakter može predstaviti kao broj), pri čemu se sumiranje vrši upotrebom funkcije za sabiranje koja je data kao argument.

- ❷ (10 bodova) Napisati funkciju *sortiraj* koja *niz* od n stringova sortira u opadajućem redoslijedu korištenjem insertion-sort algoritma, te od sortiranih stringova kreira jedan string koji vraća kao rezultat. Pri tome, parametar f predstavlja pokazivač na funkciju koja vrši spajanje dva stringa i kao rezultat vraća novokreirani string. Prototip funkcije je:

```
char* sortiraj(char **niz, int n, char * (*f)(const char *, const char *));
```

Napisati funkciju *spoji_bez_brojeva* koja spaja dva stringa ali tako da se u stringovima koji se spajaju preskaču/ignorišu brojevi. Prototip funkcije je:

```
char* spoji_bez_brojeva(const char *s1, const char *s2);
```

Napisati glavni program u kojem treba ilustrovati rad sa definisanim funkcijama tako što će se sortirati argumenti komandne linije i konačni string upisati u tekstualnu datoteku sa nazivom "izlaz.txt" koja je navedena kao poslednji argument komandne linije.

- ❸ (10 bodova) Neka je definisan tip:

```
typedef struct cvor {  
    char *podatak;  
    struct cvor *prethodni;  
    struct cvor *sljedeci;  
} CVOR;
```

kojim se reprezentuje čvor dvostruko povezane uređene liste čiji je informacioni sadržaj (dinamički) string.

Napisati funkciju koja dodaje novi string u listu tako da je poredak elemenata u listi uvijek u opadajućem redoslijedu (leksikografski). Pri tome nije dozvoljeno naknadno sortiranje. Prototip funkcije je:

```
void dodaj(CVOR **pg, CVOR **pr, char *string);
```

Napisati rekurzivnu funkciju koja računa i vraća ukupan broj pojavljivanja određenog karaktera u svim čvorovima liste. Prototip funkcije je:

```
int prebroji(CVOR *r, char karakter);
```

Parametar r predstavlja rep dvostruko povezane lliste.

- ❹ (10 bodova) Napisati program u kojem se kreira graf (ulančana reprezentacija) i vrši obilazak istog. Informacioni sadržaj grafa je pokazivač na red (sekvencijalna reprezentacija). Informacioni sadržaj reda je broj. Napisati funkciju koja omogućava obilazak grafa po širini pri čemu se za svaki posjećeni čvor grafa preuzima i ispisuje sadržaj iz reda. U glavnom programu kreirati redove koji pripadaju grafu i graf te pozvati funkciju za obilazak grafa.

Napomene:

Studenti koji popravljaju K-1, rade zadatke 1 i 2.

Studenti koji popravljaju K-2, rade zadatke 3 i 4.