

---

PROGRAMIRANJE II (10.2.2017.)

---

- ❶ (25 bodova) Napisati funkciju sa promjenljivim brojem argumenata koja sortira proizvoljan broj brojeva u opadajućem redoslijedu. Za sortiranje koristiti *shell sort* algoritam. Funkcija treba da vrati pokazivač na sortirani niz koji je formiran na *heap*-u. Definirati `main()` i ilustrovati korišćenje date funkcije.

- ❷ (25 bodova) Napisati program u kojem treba iz binarne datoteke, čiji je naziv prvi argument komandne linije, učitati matricu dimenzija  $n \times m$  (dužina reda je  $n$ ), pri čemu su  $n$  i  $m$  dati u prvih 8 bajtova binarne datoteke, kao dva četvorobajtna neoznačena podatka. Kada se matrica učitava na ovaj način, njeni biti također čine matricu. Pronaći dužinu najdužeg vertikalnog niza sukcesivnih bitova setovanih na 1.

- ❸ (25 bodova) Definirati tip `STRING` kojim se string predstavlja kao jednostruko povezana lista. Svaki čvor liste ima minimalno dvije komponente: znak i pokazivač na naredni element. Kraj stringa je definisan posebnim čvorom čiji je informacioni sadržaj znak nule, a vrijednost pokazivača na naredni element null.

Implementirati funkciju

```
STRING* nastavi_str(STRING *str1, STRING *str2);
```

koja kao rezultat vraća `STRING` koji je napravljen nastavljanjem stringova koji su dati kao argumenti i iz kojih su izbačeni čvorovi čiji je informacioni sadržaj cifra.

- ❹ (25 bodova) Data je biblioteka za rad sa usmjerenim težinskim grafom, koja u zaglavlju „*graph.h*” sadrži funkciju `double getWeight(graph* gr, int x, int y)`, koja prihvata graf i indekse dva njegova čvora i vraća rastojanje od  $x$ -tog do  $y$ -tog. Napisati funkciju

```
double maxDistance(graph* gr, int x, int y);
```

koja vraća maksimalno rastojanje između  $x$ -tog i  $y$ -tog čvora, putanjom ne dužom od četiri čvora (uključujući početni i krajnji čvor). Broj čvorova grafa sadržan je kao polje u slogu `graph`. Korišćenje pomoćnih struktura podataka, kao što su stek, red, niz i matrica, nije dozvoljeno. Nije dozvoljena alokacija dodatne dinamičke memorije.

---

PROGRAMIRANJE II (10.2.2017.)

---

- ❶ (25 bodova) Napisati funkciju sa promjenljivim brojem argumenata koja sortira proizvoljan broj brojeva u opadajućem redoslijedu. Za sortiranje koristiti *shell sort* algoritam. Funkcija treba da vrati pokazivač na sortirani niz koji je formiran na *heap*-u. Definirati `main()` i ilustrovati korišćenje date funkcije.

- ❷ (25 bodova) Napisati program u kojem treba iz binarne datoteke, čiji je naziv prvi argument komandne linije, učitati matricu dimenzija  $n \times m$  (dužina reda je  $n$ ), pri čemu su  $n$  i  $m$  dati u prvih 8 bajtova binarne datoteke, kao dva četvorobajtna neoznačena podatka. Kada se matrica učitava na ovaj način, njeni biti također čine matricu. Pronaći dužinu najdužeg vertikalnog niza sukcesivnih bitova setovanih na 1.

- ❸ (25 bodova) Definirati tip `STRING` kojim se string predstavlja kao jednostruko povezana lista. Svaki čvor liste ima minimalno dvije komponente: znak i pokazivač na naredni element. Kraj stringa je definisan posebnim čvorom čiji je informacioni sadržaj znak nule, a vrijednost pokazivača na naredni element null.

Implementirati funkciju

```
STRING* nastavi_str(STRING *str1, STRING *str2);
```

koja kao rezultat vraća `STRING` koji je napravljen nastavljanjem stringova koji su dati kao argumenti i iz kojih su izbačeni čvorovi čiji je informacioni sadržaj cifra.

- ❹ (25 bodova) Data je biblioteka za rad sa usmjerenim težinskim grafom, koja u zaglavlju „*graph.h*” sadrži funkciju `double getWeight(graph* gr, int x, int y)`, koja prihvata graf i indekse dva njegova čvora i vraća rastojanje od  $x$ -tog do  $y$ -tog. Napisati funkciju

```
double maxDistance(graph* gr, int x, int y);
```

koja vraća maksimalno rastojanje između  $x$ -tog i  $y$ -tog čvora, putanjom ne dužom od četiri čvora (uključujući početni i krajnji čvor). Broj čvorova grafa sadržan je kao polje u slogu `graph`. Korišćenje pomoćnih struktura podataka, kao što su stek, red, niz i matrica, nije dozvoljeno. Nije dozvoljena alokacija dodatne dinamičke memorije.