Fakultet tehničkih nauka, Novi Sad
Animacija u inženjerstvu

25. V 2019. godine

Diskr. i komb. met. za rač. gr.

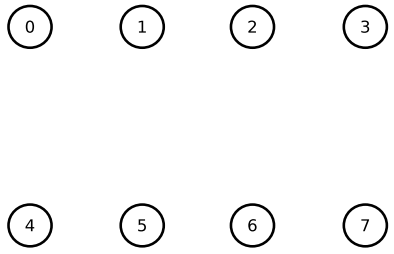
1→15, 2→15, 3→10, 4→5, 5→15, 6→10, 7→10.

1. Napisati algoritam INSERTION SORT za sortiranje umetanjem.
2. Linijama algoritma $1, 2, 3, \ldots$ iz zadatka 1. dodeliti redom vreme izvršavanja $c_1, c_2, c_3, \ldots$ Za ulazni niz dužine n koji je worst case (najgori slučaj) za INSERTION SORT izraziti ukupno vreme izvršavanja $T(n)$ preko $c_1, c_2, c_3, \ldots$
3. Dati definiciju malog o ponašanja i pokazati da je $\sqrt{n} = o(n)$. Da li za nenegativne f i g
   $f = o(g) \Rightarrow f = O(g)$?
   $f = O(g) \Rightarrow f = o(g)$?
   $f = O(g) \Rightarrow f = \Theta(g)$?
   $f = \Theta(g) \Rightarrow f = O(g)$?

---

4. Nacrtati usmereni graf G koji je dat tabelom listi susedstva:

u	Adj(u)
0	1, 4
1	2, 6
2	3, 7
3	7
4	1
5	4
6	2
7	3, 6



5. Na graf G primeniti DFS algoritam, kod čvorova napisati d i f vrednosti, kod grana napisati tip (TBCF), napraviti tabelu zagrada. Ako je dati graf DAG, dati topološko sortiranje čvorova, ako nije, dati graf komponenti jake povezanosti.
6. Napisati kod funkcije enqueue_list koja unosi čvor na kraj liste susedstva grafa. Napisati deo koda za unos grafa G (iz zad. 5) u okviru procedure main u niz listi susedstva grafa G leksikografski.

---

7. U tabeli su date cene prevoza između 5 gradova.
   (a) Polazeći od čvora 1, metodom najjeftinijeg suseda naći približno rešenje problema trg. putnika (TSP).
   (b) Za isti problem naći Mađarskom metodom angažovanje koje je rešenje relaksiranog TSP. Komentarisati rešenja (a) i (b).

	1	2	3	4	5
1	-	7	12	13	8
2	8	-	4	9	9
3	14	5	-	10	6
4	12	8	12	-	7
5	5	9	7	6	-

```
#define max_cv 50
typedef struct _node gnode;
typedef gnode *grana;
struct _node
{
    int data;
    gnode *next;
};

void enqueue_list(grana **gp, cvor d)
{
// Ovaj kod napisati
}
```

```
int main(void)
{   grana G[max_cv],GT[max_cv];
    int i,n; grana *rear[max_cv];

    for(i=0;i<max_cv;i++){
        G[i] = NULL;
        rear[i] = &(G[i]);
    }
    enqueue_list(&rear[0],1);
    // nastaviti dalje,
    // uneti ostatak grafa
    return 0;
}
```