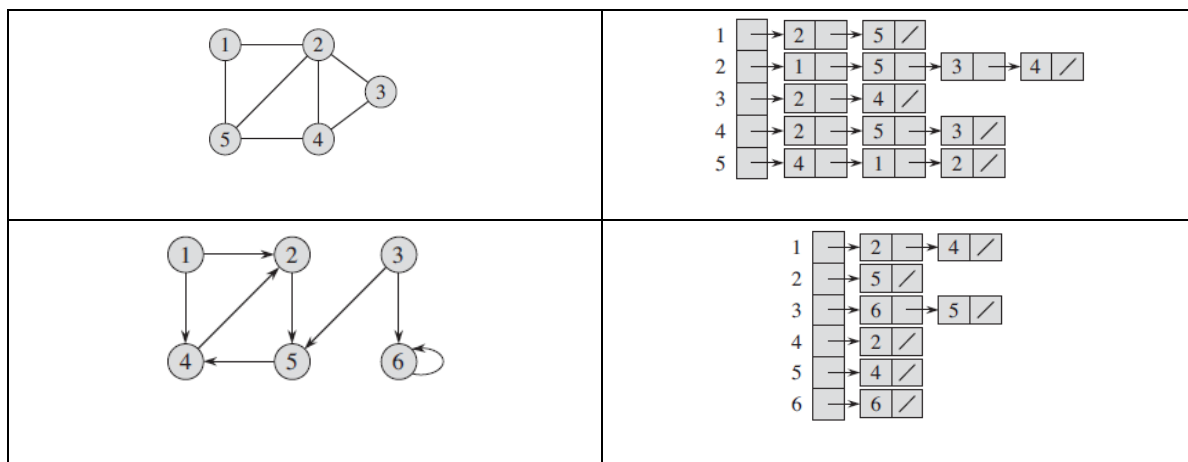


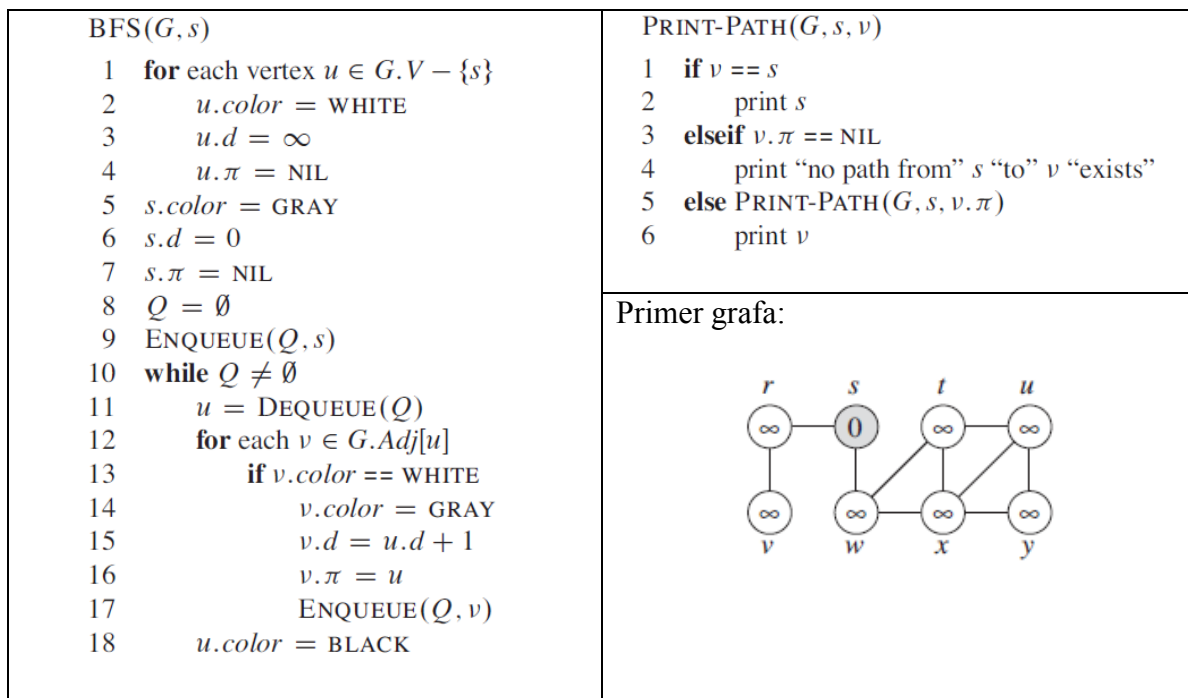
Zadaci

- Implementirati grafove prikazane na slici 1. Graf predstaviti korišćenjem reprezentacije susednih lista čvorova (*adjacency-list representation*) koja je takođe prikazana na slici 1. Čvor grafa predstaviti po uzoru na priloženi šablon. Napisati funkcije za (i) ispis svih susednih čvorova v čvora u i (ii) ispis svih veza čvora u i susednih čvorova v .



Slika 1 – Ilustracija usmerenog i neusmerenog grafa (levo) i implementacija istih upotrebom listi (desno).

- Implementirati pretragu grafa po širini (*Breadth-first search*). Za proveru funkcionalnosti iskoristiti graf sa slike 2. Takođe, na slici 2 dati su pseudokodovi funkcija.



Slika 2 – Pseudokod funkcija za pretragu grafa po širini.

3. Implementirati pretragu grafa po dubini (*Depth-first search*). Za proveru funkcionalnosti iskoristiti graf sa slike 3. Takođe, na slici 3 dati su pseudokodovi funkcija.

<pre> DFS(G) 1 for each vertex $u \in G.V$ 2 $u.color = WHITE$ 3 $u.\pi = NIL$ 4 $time = 0$ 5 for each vertex $u \in G.V$ 6 if $u.color == WHITE$ 7 DFS-VISIT(G, u) </pre>	<pre> DFS-VISIT(G, u) 1 // white vertex u has just been discovered 1 $time = time + 1$ 2 $u.d = time$ 3 $u.color = GRAY$ 4 // explore edge (u, v) 4 for each $v \in G.Adj[u]$ 5 if $v.color == WHITE$ 6 $v.\pi = u$ 7 DFS-VISIT(G, v) 8 // blacken u; it is finished 8 $u.color = BLACK$ 9 $time = time + 1$ 10 $u.f = time$ </pre>
<p>Primer grafa:</p>	

Slika 3 – Pseudokod funkcija za pretragu grafa po dubini.

4. Koristeći pretragu grafa po dubini implementirati topološki sort algoritam čiji je pseudokod prikazan na slici 4. Testni i rezultatni graf je takođe prikzani na slici 4.

<pre> TOPOLOGICAL-SORT(G) 1 call DFS(G) to compute finishing times $v.f$ for each vertex v 2 as each vertex is finished, insert it onto the front of a linked list 3 return the linked list of vertices </pre>
<p>Ulazni graf:</p>
<p>Izlazni graf:</p>

Slika 4 – Pseudokod funkcija topološkog sorta i primer testnog i rezultatnog grafa.