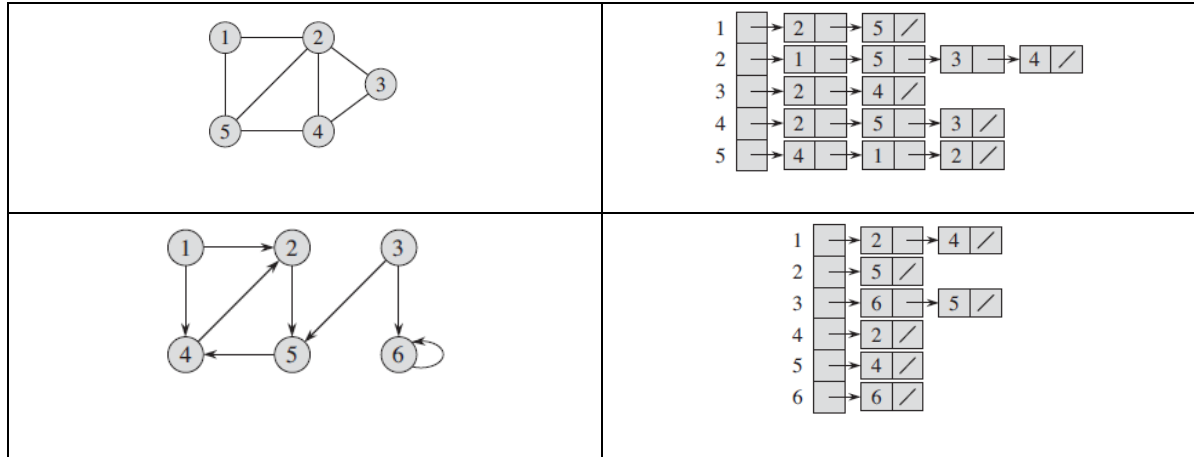


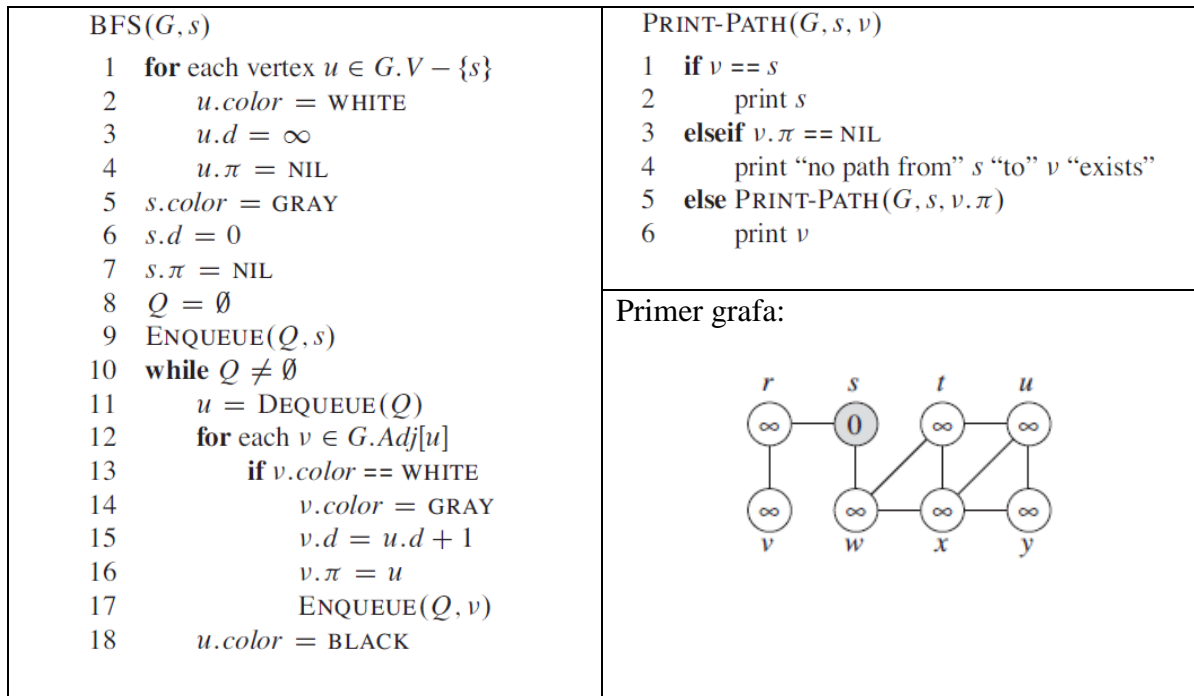
## Zadaci

- 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  $v$ .



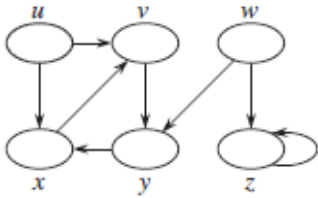
Slika 1 – Ilustracija usmerenog i neusmerenog grafa (levo) i implementacija istih upotrebom listi (desno).

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

Slika 3 – Pseudokod funkcija za pretragu grafa po dubini.