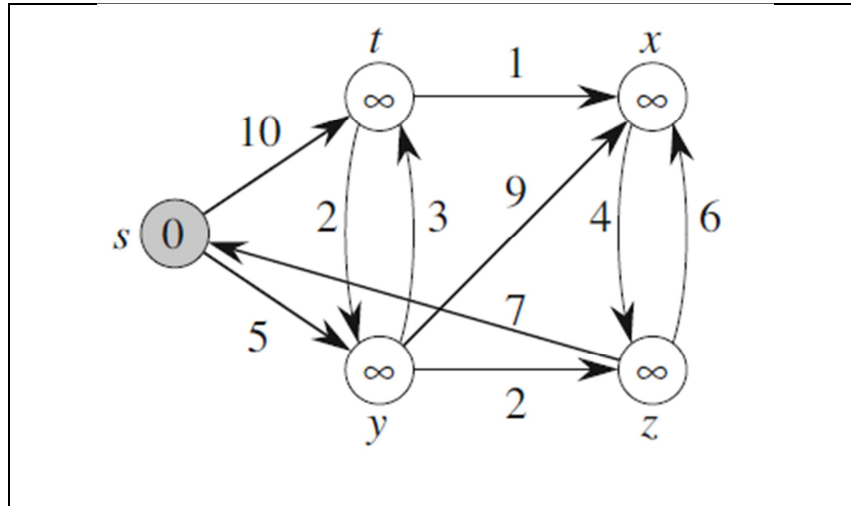


Zadaci

- Implementirati graf prikazan na slici 1. Izmeniti klasu *Vertex* tako da podrži definisanje težinskih koeficijenata na ivicama grafa. Pored klase *Vertex* napraviti i klasu *Edge* koja predstavlja ivicu grafa. Klasa *Edge* treba da sadrži barem *source*, *destination* i *weight* atribute.

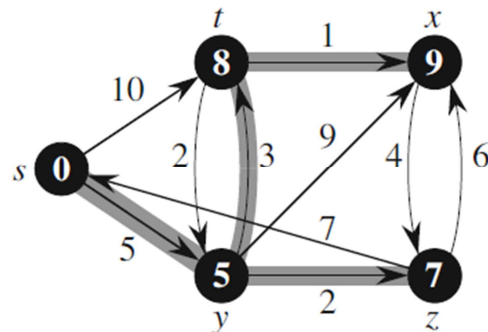


Slika 1 – Ilustracija usmerenog težinskog grafa.

- Implementirati Dijkstrin algoritam za pronalaženje najkraće putanje grafa. Pseudokodovi potrebnih funkcija su dati na slici 2. Kao ulazni graf koristiti graf sa slike 1. Na slici 2 je takođe prikazan i rezultujući graf.

<pre> INITIALIZE-SINGLE-SOURCE(G, s) 1 for each vertex $v \in G.V$ 2 $v.d = \infty$ 3 $v.\pi = \text{NIL}$ 4 $s.d = 0$ </pre>	<pre> DIJKSTRA(G, w, s) 1 INITIALIZE-SINGLE-SOURCE(G, s) 2 $S = \emptyset$ 3 $Q = G.V$ 4 while $Q \neq \emptyset$ 5 $u = \text{EXTRACT-MIN}(Q)$ 6 $S = S \cup \{u\}$ 7 for each vertex $v \in G.Adj[u]$ 8 RELAX(u, v, w) </pre>
<pre> RELAX(u, v, w) 1 if $v.d > u.d + w(u, v)$ 2 $v.d = u.d + w(u, v)$ 3 $v.\pi = u$ </pre>	

Rezultat algoritma:



Slika 2 – Pseudokod funkcija potrebnih za implementaciju Dijkstrinog algoritma.

3. Napraviti funkciju za nasumično generisanje usmerenog težinskog grafa. Korišćenjem funkcije *PrintPath* (Vežba 7) ispisati sve putanje od čvora u ka svim ostalim čvorovima, i na osnovu toga skicirati dobijeni graf. Čvor u takođe zabrati nasumično.