

1. Zadana je lista **a** = [5, 9, 3, 1, 6, 4, 2].

- (5) nacrtajte binarno stablo traženja u čijim će se čvorovima nalaziti redom elementi zadane liste **a**.
- (1) ispišite korijen nacrtanog stabla.
- (1) koji je čvor roditelj čvora 6?
- (2) ispišite sve listove nacrtanog stabla.
- (5) zadana je klasa **Tree**:

```
class Tree:
    def __init__(self, v, l=None, r=None):
        self.value = v
        self.left = l
        self.right = r
```

Napišite naredbe koje će rezultirati da u varijabli **root** bude pokazivač na korijen cijelog binarnog stabla iz a) dijela zadatka.

- (2) koja je vremenska složenost dodavanja čvora u binarno stablo traženja?
- (4) napišite metodu **postorder(self)** koja će vraćati listu, koja predstavlja postorder obilazak binarnog stabla.
- (2) navedite bar jednu vrstu balansiranog binarnog stabla traženja (binarno stablo traženja koje je skoro potpuno).
- (10) napišite metodu **najmanji(self)** koja čvora binarnog stabla.

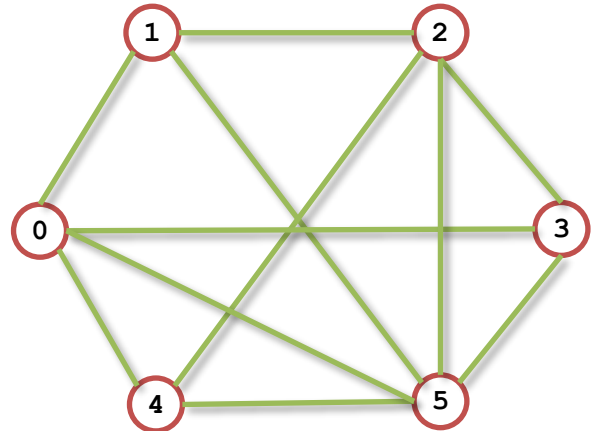
2. Zadana je hash funkcija **h(k) = k % 9** te hash tablica sa 9 pretinaca koji su redom indeksirani brojevima od 0 do 8. U koje pretince u tablici će biti dodavani sljedeći elementi ako se sudari rješavaju otvorenim adresiranjem s prvim slobodnim mjestom u tablici.

- (1) element 2
- (1) element 20
- (1) element 29

0	
1	
2	
3	
4	
5	
6	
7	
8	

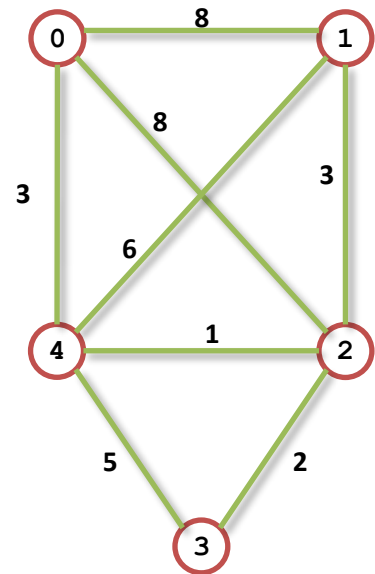
3. Za graf na slici odredite:

- (2) skup čvorova grafa.
- (2) napišite bar dva ciklusa duljine 4.
- (3) matricu susjedstva grafa.
- (3) listu susjedstva grafa.
- (5) ispišite čvorove grafa onim redoslijedom kako ih obilazimo dubinskim (DFS) obilaskom krenuvši od čvora 0 (pri odabiru susjeda nekog čvora uvijek biramo čvorove redom od onog s najmanjom vrijednosti prema onom s najvećom vrijednosti).
- (10) napišite funkciju čiji će parametar biti matrica susjedstva grafa, a vraćati će *True* ako u grafu postoji izolirani čvor, a *False* inače.



4. Za graf na slici:

- (5) opišite postupak kreiranja maksimalnog razapinjućeg stabla koristeći Kruskalov algoritam.
- (2) u koju strategiju rješavanja problema možemo uvrstiti Kruskalov algoritam i zašto?
- (3) napišite težinsku matricu grafa na slici.
- (10) Napišite funkciju čiji će parametar biti težinska matrica usmjerenog grafa, a vraćati će najkraću udaljenost neka dva direktno povezana čvora.



5. Za neusmjereni graf reći ćemo da je *dvoobojiv* ako se čvorovi grafa mogu obojati s dvije boje na način da nikoja dva vrha koji su direktno spojeni nisu obojeni istom bojom.

- (5) Koji algoritam biste mogli iskoristiti za provjeru je li graf dvoobojiv (obrazložite svoj odgovor)?
- (15) Napišite funkciju `dvoobojiv(a)` koja će za zadanu matricu susjedstva grafa (*a*) vraćati *True* ako je graf dvoobojiv, a *False* inače.