

*Vežbe br. 5*

# GRAFOVI

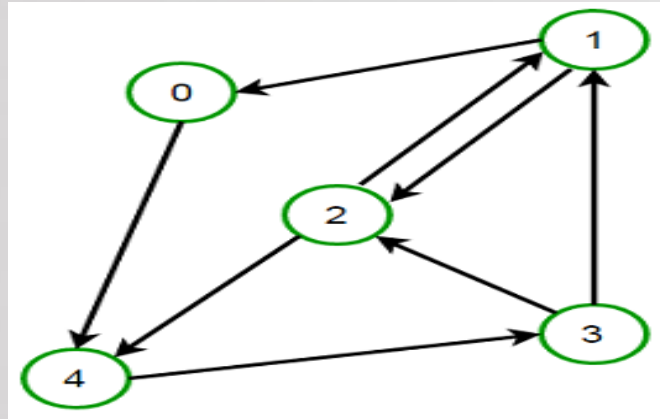
(deo 3 – zadaci za vežbu)

# ZADACI ZA VEŽBU

## Zadatak 1:

Modifikovati DFS algoritam tako da vrši proveru da li je graf „jako povezan“. Graf je „jako povezan“ ako se iz svakog čvora može stići do svakog drugog čvora u grafu.

Primer. Graf sa slike ispod je „jako povezan“.

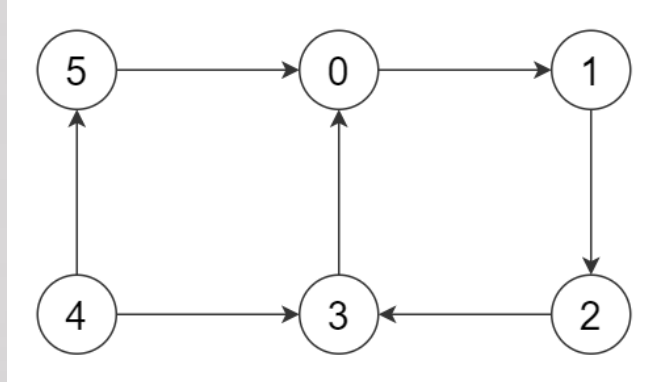


# ZADACI ZA VEŽBU

## Zadatak 2:

Modifikovati DFS algoritam tako da vraća indeks „izvornog“ čvora u grafu. Izvorni čvor je onaj čvor od kog se može stići do svakog drugog čvora u grafu.

Primer. U grafu sa slike ispod izvorni čvor je čvor 4.



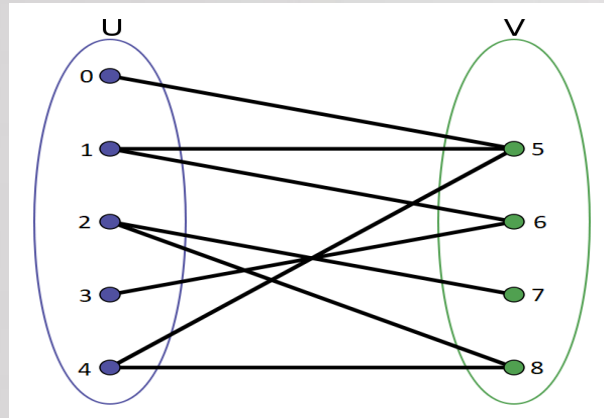


# ZADACI ZA VEŽBU

## Zadatak 3:

Za dati neusmereni graf izvršiti proveru da li je bipartitan. Bipartitan graf je onaj kod kog se čvorovi mogu podeliti u dve grupe, pri čemu je svaki čvor iz jedne grupe povezan sa bar jednim iz druge grupe, dok čvorovi iz iste grupe ne formiraju veze.

Primer: Graf sa slike ispod je bipartitan.



```
class Graph:
```

```
    def __init__(self):  
        self.graph = {}
```

```
    def addEdge(self, u, v):  
        if u not in self.graph:  
            self.graph[u] = []  
        if v not in self.graph:  
            self.graph[v] = []  
        self.graph[u].append(v)
```

```
    def DFS(self, v, visited):  
        visited[v] = True
```

```
        for u in self.graph[v]:  
            if not visited[u]:  
                self.DFS(u, visited)
```

**KRAJ** 😊