

## SMA Lab Assignment 1: Exploring networks using networkx package

**SURIYA S 225229140**

```
In [1]: import networkx as nx
```

```
In [2]: G = nx.Graph()
```

```
In [3]: print(G.nodes())  
print(G.edges())
```

```
[]  
[]
```

```
In [4]: G.add_node("A")
```

```
In [5]: G.add_nodes_from(["B", "C", "D", "E"])
```

```
In [6]: G.add_edge(*("A", "B"))
```

```
In [7]: G.add_edges_from([("A", "C"), ("B", "D"), ("B", "E"), ("C", "E")])
```

### Accessing vertex and edge sets

```
In [8]: print("Vertex set: ", G.nodes())
```

```
Vertex set:  ['A', 'B', 'C', 'D', 'E']
```

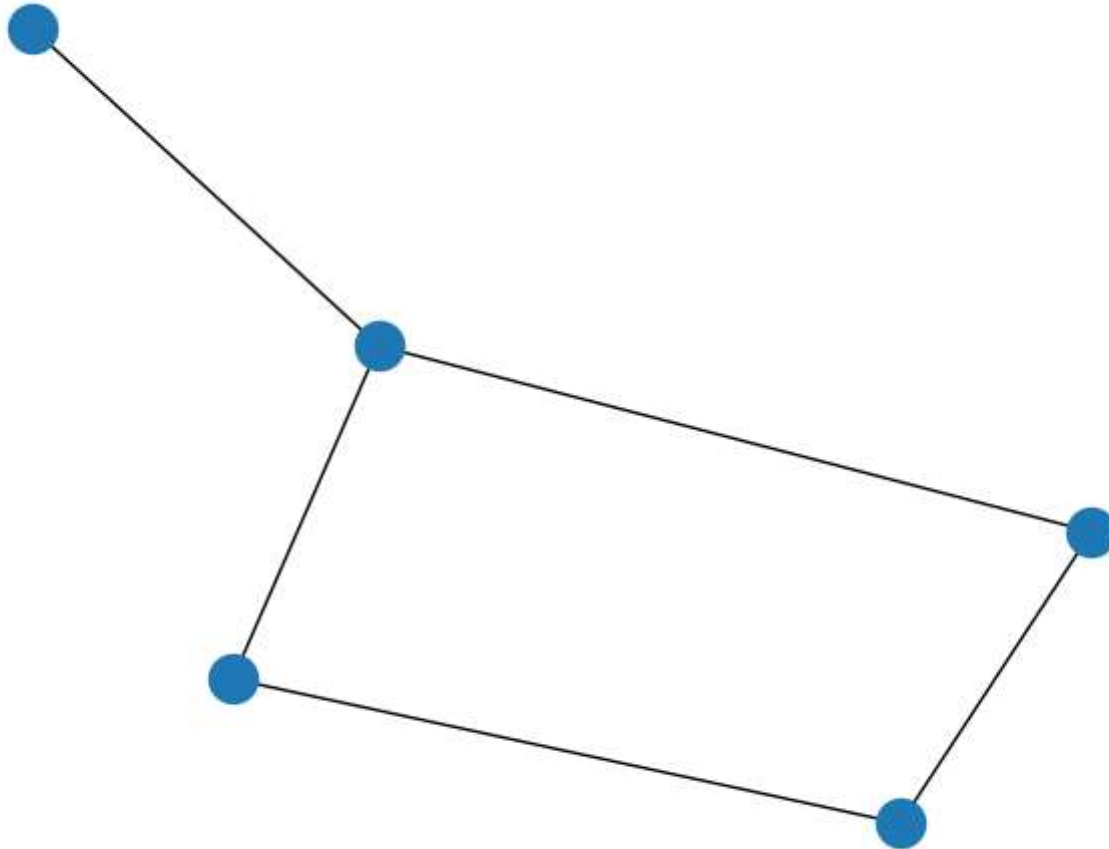
```
In [9]: print("Edge set: ",G.edges())
```

```
Edge set:  [('A', 'B'), ('A', 'C'), ('B', 'D'), ('B', 'E'), ('C', 'E')]
```

### Drawing graph

```
In [10]: import matplotlib.pyplot as plt
```

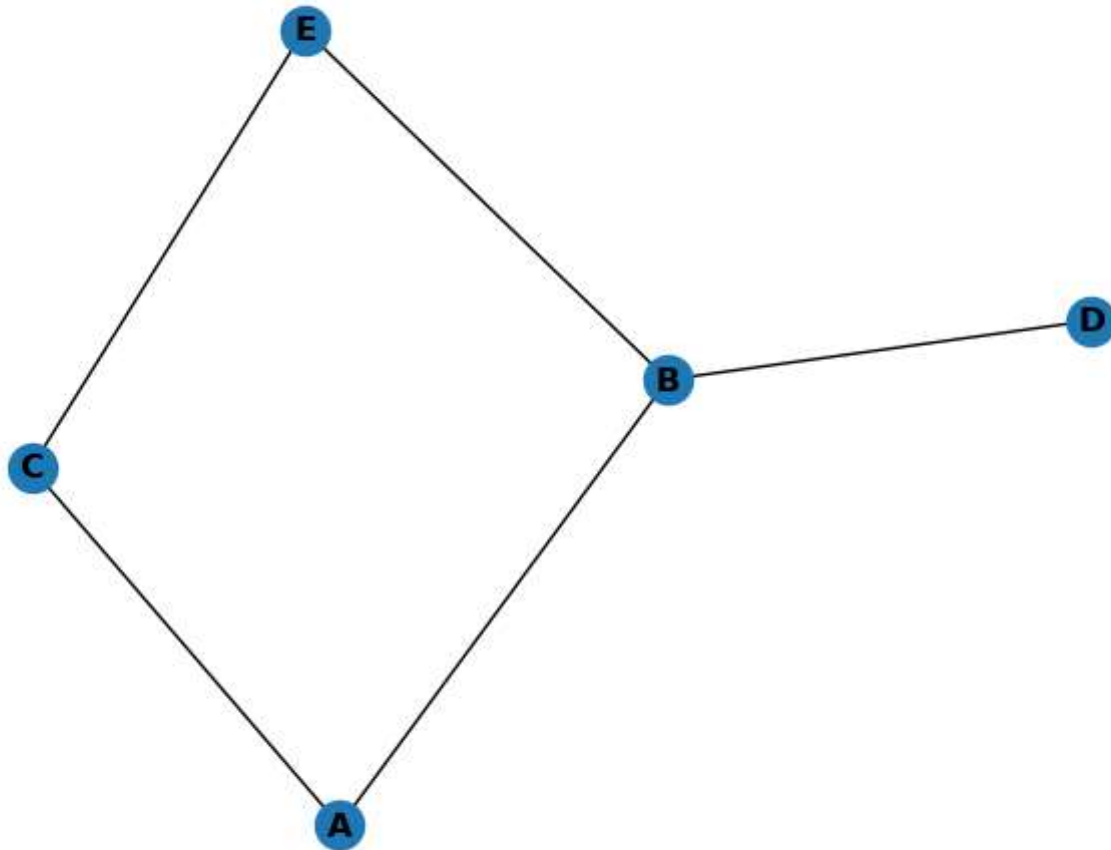
```
In [11]: nx.draw(G)  
plt.show()
```



```
In [12]: plt.savefig("graph.png")
```

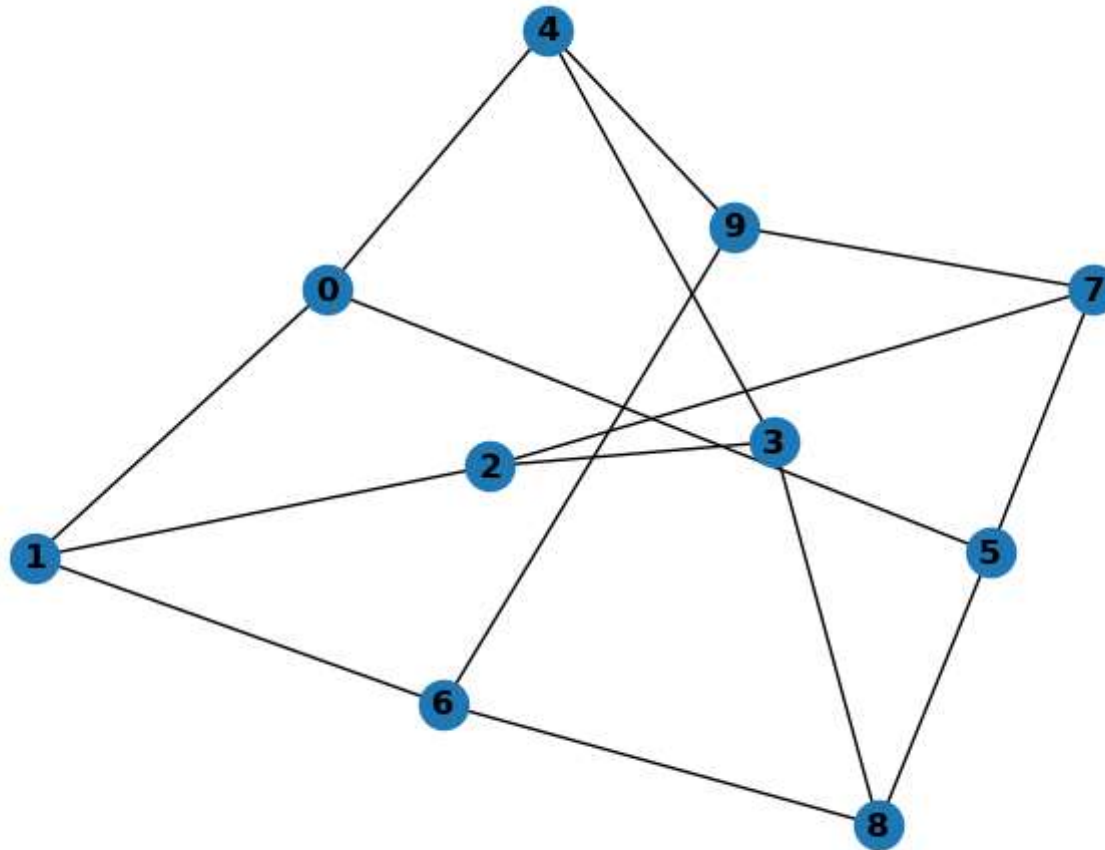
<Figure size 640x480 with 0 Axes>

```
In [13]: nx.draw(G, with_labels=True, font_weight='bold')  
plt.show()
```



```
In [14]: GP = nx.petersen_graph()
```

```
In [15]: nx.draw(GP, with_labels=True, font_weight='bold')
plt.show()
```



#### Adjacency view

```
In [16]: print(G.adj)
```

```
{'A': {'B': {}, 'C': {}}, 'B': {'A': {}, 'D': {}, 'E': {}}, 'C': {'A': {}, 'E': {}}, 'D': {'B': {}}, 'E': {'B': {}, 'C': {}}}
```

### Degree of a vertex

```
In [17]: G.degree("A")
```

```
Out[17]: 2
```

### Creating weighted graph

```
In [18]: G = nx.Graph()  
E = [('A', 'B', 2), ('A', 'C', 1), ('B', 'D', 5), ('B', 'E', 3), ('C', 'E', 2)]  
G.add_weighted_edges_from(E)
```

```
In [19]: pos=nx.spring_layout(G)
nx.draw(G, pos, with_labels=True, font_weight='bold')
edge_weight = nx.get_edge_attributes(G,'weight')
nx.draw_networkx_edge_labels(G, pos, edge_labels = edge_weight)
plt.show()
```

