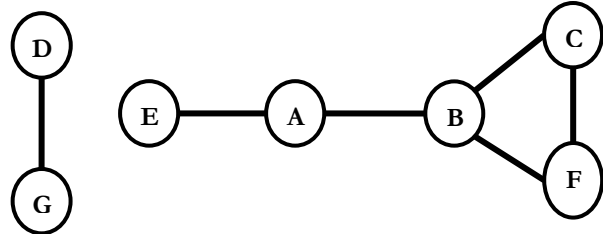


Exercise Series & Lab N°05: Graphs and Elementary Graph Algorithms

Exercise V.1 (Graph Properties)

A) Consider the following graph $G = (V, E)$.



Q1) What is the type of graph G ?

Q2) Find V , E , $|V|$, and $|E|$.

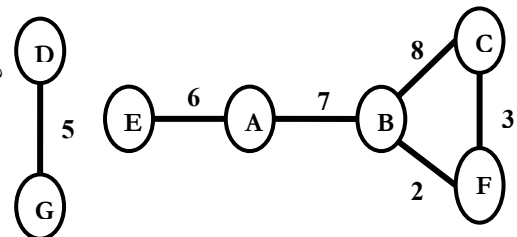
Q3) What is the maximum degree of the graph?

Q4) Are there any cycles? If so, where?

Q5) What is the maximum length simple path in this graph?

Q6) What is one edge you could add to the graph that would increase the length of the maximum length simple path of the new graph to **6**?

Q7) What are the connected components of the graph?



B) Consider the following graph $G' = (V', E')$.

Q8) What is the type of graph G' ?

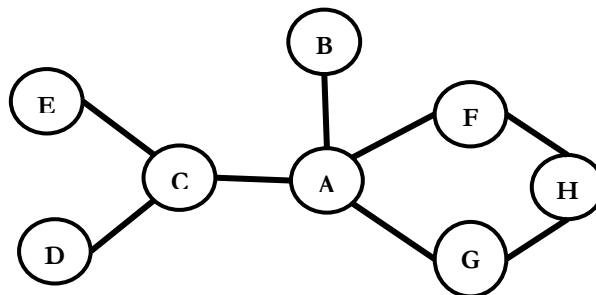
Q9) What is the path involving the least number of nodes from E to C ? What is its cost?

Q10) What is the minimum cost path from E to C ? What is its cost?

Q11) What is the minimum length path from E to C ? What is its length?

Exercise V.2 (Graph Traversal)

Consider the following graph. Suppose we want to traverse it, starting at node A .

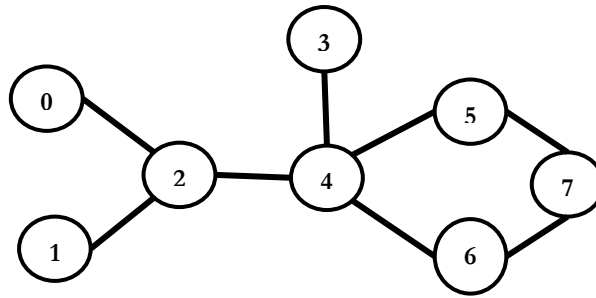


Q1) If we traverse this using **breadth-first search (BFS)**, what are **two** possible orderings of the nodes we visit?

Q2) If we traverse this using **depth-first search (DFS)**, what are **two** possible orderings of the nodes we visit?

Exercise V.3 (Adjacency Matrix)

Consider the following graph $G = (V, E)$.



- Q1) Write the structure of the graph to represent it using the adjacency matrix.
- Q2) Write a **Create_edges_AM()** function to create all the edges to the graph.
- Q3) Write an **Init_Adj_Matrix()** function to initialize all adjacency Matrix values to zero.
- Q4) Write an **Add_edge_AM()** function to create all the edges to the graph
- Q5) Write a **Create_Adj_Matrix ()** function to create Adjacency Matrix to the graph.
- Q6) Write a **Print_Adj_Matrix ()** function to print the created Adjacency Matrix.

Exercise V.4 (Adjacency List)

Consider the graph $G = (V, E)$ presented in **Exercise V.3**.

- Q1) Write the structure of the graph to represent it using the adjacency list.
- Q2) Write a **Create_GNode ()** function to create a new graph node.
- Q3) Write a **Create_AL_Graph ()** function to create an Adjacency List to the graph.
- Q4) Write an **Add_Edge_AL()** function to add an edge to the graph
- Q5) Write a **Create_Edges_AL()** function to create all the edges to the graph.
- Q6) Write a **Print_AL_Graph ()** function to print the Adjacency List of the graph.

Exercise V.5 (Homework : DFS & BFS)

- Q1) Create the graph $G = (V, E)$ shown in **Exercise V.3** using an Adjacency List.
- Q2) Write a **DFS () iterative** function to traverse the graph using **depth -first search (DFS)**.
- Q3) Write a **BFS ()** function to traverse the graph using **breadth-first search (BFS)**.