|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Part** | **1** | **2** | **3** | **4** | **Total** |
| *maximum* | **25** points | **25** points | **25** points | **25** points | **100**G101010 pointsG |
| ***Your Score*** |  |  |  |  |  |

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Sets and Dictionaries**

Reading Assignment: Thoroughly read Chapter 12 in the course textbook.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Part 1 Glossary Terms**

Define, in detail, each of these glossary terms from the realm of computer programming logic and design and computer topics, in general. If applicable, use examples to support your definitions. Consult your notes or course textbook(s) as references or by visiting Web sites such as: [**http://www.ask.com**](http://www.ask.com),[**http://www.bing.com**](http://www.bing.com), [**http://www.webopedia.com**](http://www.webopedia.com)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**(a) Adjacency Matrix**

|  |
| --- |
| An adjacency matrix is used to represent a graph. There are two implementations, one is a Grid or Matrix and the other is an Array of Linked Lists.  In the grid or matrix implementation, the intersection of rows and columns represent edges, and a 0 or 1 represent whether an edge exists or not. This is better for denser graphs.  In a linked list implemented, an array contains a linked list for each vertex, and links to nodes that it has edges to. This is better for sparse graphs, and has faster insertions and has less memory usage if the graph isn’t too dense. |

**(b) Complete Graph**

|  |
| --- |
| Each vertex is a complete graph has an edge to each other vertex in the graph. |

**(c) Connected Graph**

|  |
| --- |
| A connected graph has a path that connects each vertex in a graph. |

**(d) Directed Graph**

|  |
| --- |
| A directed graph (digraph) has edges that indicate direction. |

**(e) Weighted Graph**

|  |
| --- |
| A weighted graph has edges with weight values, which can indicate such things as e.g. sky miles in a flight graph. The shortest possible paths in a weighted graph is the minimum sum of the weights from each edge in a path from a source vertex to all the other connected vertices. Djikstra’s algorithm (implemented as an exercise in Chapter 12) solves this problem with O(n^2) complexity. |

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Part 2 True / False Exercises**

For each of these exercises, enter True or False in the spaces provided.

**TRUE (1)** An example of a process or model that can be graphed is the links between pages on the Internet.

**FALSE (2)** On a weighted graph, the vertices are labeled with numbers. **EDGES ARE WEIGHTED WITH NUMBERS**

**FALSE** **(3)** In a connected graph, there must be an edge from each vertex to every other vertex.

**TRUE (4)** In a complete graph with six vertices, the degree of a vertex is five.

**TRUE** **(5)** The adjacency matrix representation of a graph stores graph information in an array of lists.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Part 3 Multiple Choice Exercises**

Select the correct response or responses.

**(1)** Which of the following is NOT a process for which a graph can serve as a model?

(a) a road map between hotels a town

**(b) a line at a movie theater**

(c) the paths that data can travel in a network

(d) the routes between rooms in a building

**(2)** Which of the following is true about graphs?

(a) graphs consist of vertices and nodes

(b) the edges between vertices are always labeled

(c) an adjacency is when one vertex has a path to another vertex

**(d) the length of a path is the number of edges on the path**

**(3)** What makes a graph complete?

**(a) when there is an edge from each vertex to all other vertices**

(b) when there is a path from each vertex to all other vertices

(c) when there is a path between at least half the vertices

(d) when there are two or more edges between vertices

**(4)** Which term best describes a neighbor?

(a) a path exists between vertices

(b) a vertex is reachable from another vertex

(c) two vertices have consecutive labels

**(d) two vertices are adjacent**

**(5)** The number of edges connected to a vertex describes which of the following?

(a) a complete graph

(b) the neighbor of a vertex

**(c) the degree of a vertex**

(d) whether a graph is connected

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

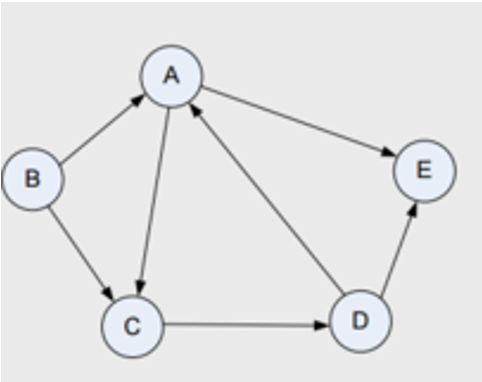
**Part 4 Programming Exercises**

**(1)** In a component with *n* vertices, how many edges are in the spanning tree?

**N – 1 edges**

**(2)** Review the directed graph and then complete its associated adjacency matrix.

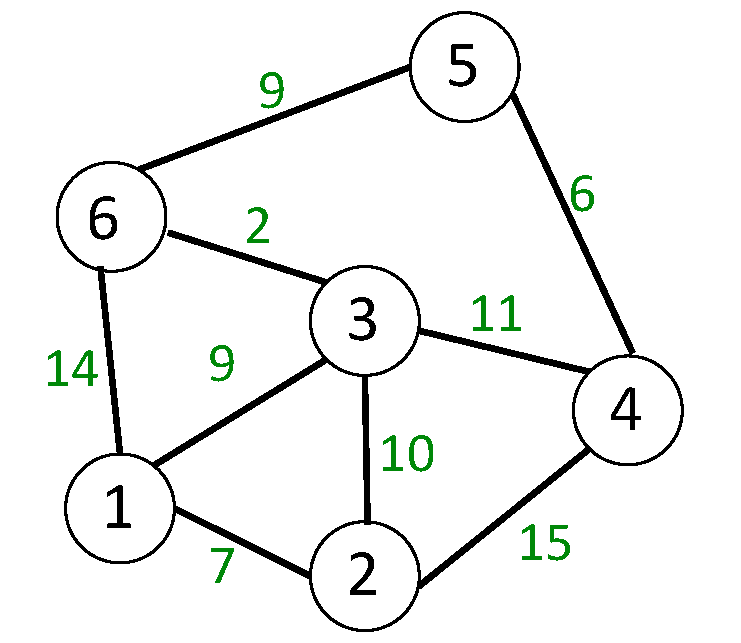
**Directed Graph**

****

**Adjacency Matrix**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **A** | **B** | **C** | **D** | **E** |
| **A** | **0** | **0** | **1** | **1** | **1** |
| **B** | **1** | **0** | **1** | **0** | **0** |
| **C** | **0** | **0** | **0** | **1** | **0** |
| **D** | **1** | **0** | **0** | **0** | **1** |
| **E** | **0** | **0** | **0** | **0** | **0** |

**(3)** Examine the given weighted graph. Consider a tour of the vertices of the graph. What is the shortest path from Vertex 1 to Vertex 5 ?  **1-> 3 -> 6 -> 5**

****

**(4) ( Representing Graphs )**

A weighted graph is represented by a multi - dimensional list, as shown in this program segment. Here each row shows two vertices and a numerical weight.

The first element in the list is **["C", "A", 3]** which indicates that Vertex C is connected to Vertex A and the edge connecting them has a weight of 3 .

Using a hand - drawing or a digital drawing tool, attempt to re - create the graph to appear as is typical of a graph in data structures.

|  |
| --- |
| **print()**  **graph = [["C", "A", 3], ["A", "C", 4], ["B", "C", 2],**  **["A", "B", 5], ["C", "B", 8], ["C", "D", 4]]**  **print (graph)**  **print()**  **for i in range(len(graph)) :**  **for j in range(len(graph[i])) :**  **print(graph[i][j], end = " ")**  **print()**  **print()** |

**Printing the following graph using my own class functions:**

g = LinkedDirectedGraph(["A", "B", "C", "D", "E"])

edgeList = [["C", "A", 3], ["A", "C", 4], ["B", "C", 2],

["A", "B", 5], ["C", "B", 8], ["C", "D", 4]]

*for* edge *in* edgeList:

g.addEdge(edge[0], edge[1], edge[2])

g.printGraph()

test: LinkedVertex = g.getVertex("A")

shortestPath = g.shortestPaths(test)

print(shortestPath["title"])

print(shortestPath["table"])

**RETURNS   
  
A: (A —> C) (A —> B)**

**B: (B —> C)**

**C: (C —> A) (C —> B) (C —> D)**

**D:**

**E:**

**Set of shortest paths for origin A**

**| Index | Vertex | Distance | Parent |**

**|---------|----------|------------|----------|**

**| 0 | A | 0 | |**

**| 1 | B | 5 | A |**

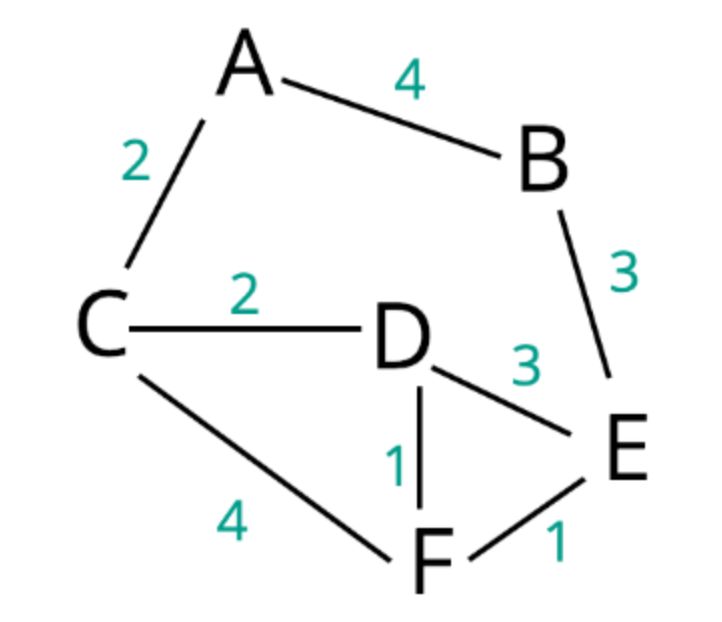
**| 2 | C | 4 | A |**

**| 3 | D | 8 | C |**

**| 4 | E | 0 | |**

**(5) ( Dijkstra's Algorithm )**

Find the shortest path from Vertex A to Vertex E .

****

**A>B>E**