DSC 40B - Midterm 02 Review

Problem 1.

The goal of contact tracing is to determine how the spread of a virus occurs. Which type of graph would be best for modelling the spread of a virus?

☐ Directed graph

☐ Undirected graph

Solution: Directed graph

Problem 2.

A directed graph has 7 nodes. What is the maximum number of edges it can have?

Solution: 49

Problem 3.

An undirected graph has 12 nodes. What is the maximum number of connected components it can have?

Solution: 12

Problem 4.

A directed graph has 5 nodes. What is the largest degree that a node in the graph can possibly have?

Solution: 10

Problem 5.

Assume that a hash table is implemented using chaining to resolve collisions. The hash table stores n numbers. Suppose a membership query is made for 42.
True or False: more than one bin in the hash table may be checked with a linear search during the query.

☐ True

☐ False

Solution: False

Problem 6.

Suppose a hash table in constructed so that collisions are resolved through chaining. The hash table has a number of bins k that is $\Theta(n)$, where n is the number of elements being stored in the hash table; that is, the hash table grows as more elements are inserted.

Suppose the hash function uses only the first $k/2$ bins of the hash table, but appears to hash items into these bins evenly. What is the expected time complexity of a membership query on this hash table? State your answer in asymptotic notation in terms of the number of elements in the hash table, n.

Solution: $\Theta(1)$ since only $k/2$ bins are used, the expected number of elements is $\frac{n}{k/2} = \frac{2n}{k}$. But since $k = \Theta(n)$ this is just $\Theta(1)$.

Problem 7.

Both Full BFS and Full DFS can be used to count the number of connected components in an undirected graph.

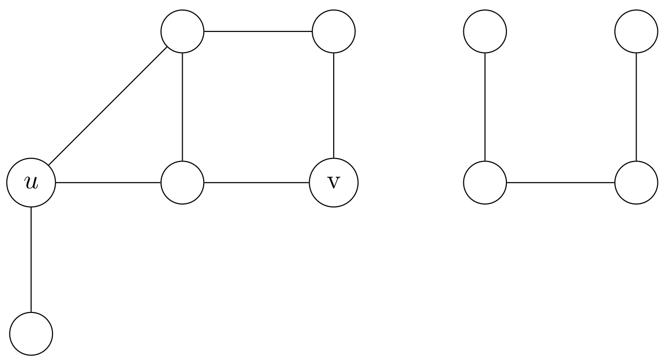
☐ True

☐ False

Solution: True

Problem 8.

How many paths are there from node u to node v in the graph below?



☐ Infinitely many

☐ 4

☐ 3

☐ 5

Solution: Infinitely many

Problem 9.

In an unweighted graph, there is at most one shortest path between any pair of given nodes.

☐ True

☐ False

Solution: False

Problem 10.

An undirected graph has 5 nodes. What is the smallest number of connected components it can have?

Solution: 1

Problem 11.

In a full BFS of a graph G=(V, E), the number of times that something is popped form the queue is 2V if the graph is undirected and V if the graph is directed.

☐ True

☐ False

Solution: False

Problem 12.

In BFS it is possible for the queue to simultaneously contain a node whose distance from the source is 3 and node whose distance from the source is 5.

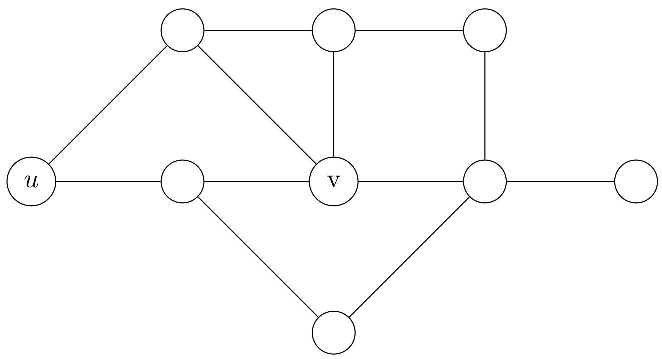
☐ True

☐ False

Solution: False

Problem 13.

Suppose a BFS is run on the graph below with u as the source.



Of course, u is the first node to be popped of the queue. Suppose that node v is the kth node popped from the queue.

a) What is the smallest that k can possibly be?

Solution: 4

b) What is the largest that k can possibly be?

Solution: 5

Problem 14.

Consider the modified BFS given below:

```
def bfs(graph, source, status=None):
    if status is None:
        status = {node: 'undiscovered' for node in graph.nodes}
```

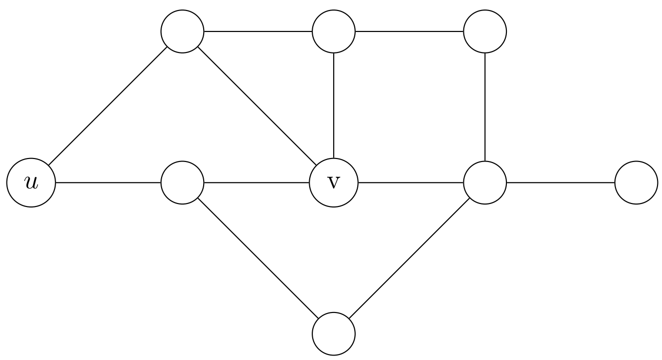
```python
    status[source] = 'pending'
    pending = deque([source])

    # while there are still pending nodes
    while pending:
        u = pending.popleft()
        for v in graph.neighbors(u):
            # explore edge (u,v)
            if status[v] == 'undiscovered':
                print ("Hey")
                status[v] = 'pending'
                # append to right
                pending.append(v)
        status[u] = 'visited'
```

Suppose this code is run on a connected undirected graph with 12 nodes. Exactly how many times will 'Hey' be printed?

Solution: 11

Problem 15.

Suppose a DFS is run on the graph below with u as the source.



Node u will be the first node marked pending. Suppose that node v is the kth node marked pending.

a) What is the smallest that k can possibly be?

Solution: 3

b) What is the largest that k can possibly be?

Solution: 9

Problem 16.

If DFS is called on a complete graph, the time complexity is $\theta(V^2)$

☐ True

☐ False

Solution: True

Problem 17.

What is the result of updating the edge (u,v) when the est[u], est[v] and weight(u,v) are given as follows?

Figure 1: Bellman Ford update subroutine

```
def update(u, v, weights, est, predecessor):
    if est[v] > est[u] + weights(u,v):
       est[v]=est[u]+weights(u,v)
       predecessor[v]=u
       return True
    else:
       return False
```

a) est[u] = 7, est[v] = 11, weight(u,v) = 3

Solution: est[v] is updated to 10

b) est[u] = 15, est[v] = 12, weight(u,v) = -3

Solution: est[v] is not updated

c) est[u] = 12, est[v] = 14, weight(u,v) = 3

Solution: est[v] is not updated

Problem 18.

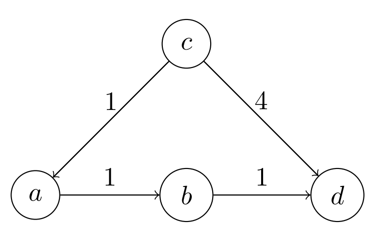
State TRUE or FALSE for the following statements:

a) If (s, v_1, v_2, v_3, v_4) is a shortest path from s to v_4 in a weighted graph, then (s, v_1, v_2, v_3) is a shortest path from s to v_3

Solution: True. Assume for the sake of contradiction that there is a path P from s to v_3 whose weight is lesser than (s, v_1, v_2, v_3). Then we can find a path from s to v_4 by combining P with the edge (v_3, v_4) whose weight is lesser than (s, v_1, v_2, v_3, v_4). This contradicts the fact that (s, v_1, v_2, v_3, v_4) is a shortest path from s to v_4.

b) Let P be a shortest path from some vertex s to some other vertex t in a directed graph. If the weight of each edge in the graph is increased by one, P will still be a shortest path from s to t.

Solution: False.



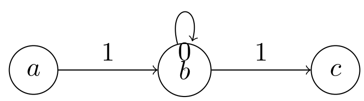
Consider the graph given above. The shortest path from c to d is (c,a,b,d) which is of weight 3. However, if the weight of each edge is increased by 1, the shortest path from c to d would be (c,d) of weight 5.

c) Suppose the update function is modified such the est[v] is updated when est[v] $\geq$ est[u] + weight(u,v) instead of strictly greater than. The est values of all nodes at the end of the algorithm would still give the shortest distance from the source.

Solution: True.

d) Suppose the update function is modified such the est[v] is updated when est[v] $\geq$ est[u] + weight(u,v) instead of strictly greater than. We can still find the shortest path from the source to any node using the predecessors using the new algorithm.

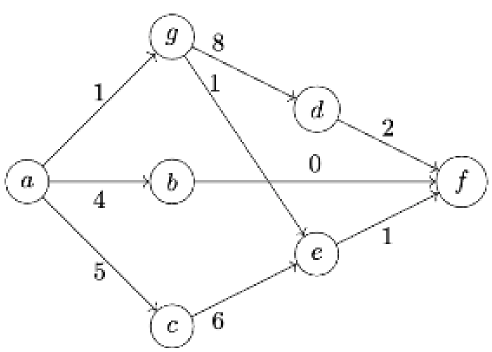
Solution: False.



Consider the graph given above. Let a be the source in the execution of Bellman Ford algorithm. Updating the edge will result in est[b] = 1 and predecessor[b] = a. However, updating the edge (b,b) will result in predecessor[b] = b according to the new update algorithm. Therefore, it would not be possible to find the shortest path from a to b using the predecessors.

Problem 19.

Suppose Dijkstra's algorithm is run on the graph shown below using node a as the source.



Suppose C is the set of "correct" nodes: that is, C is the set of nodes whose estimated distances are known to be correct.

What will be the fifth node added to C by Dijkstra's algorithm? (The first node added to C is the source, a.)

Solution: b