Name: ____________________________ **RCS ID:** ___________ @rpi.edu

CSCI 2300 — Introduction to Algorithms
Fall 2020 Exam 1 (February 20, 2020) — SOLUTIONS

- Please silence and put away all laptops, phones, calculators, electronic devices, etc.
- You may use your printed notes and book(s) for this exam
- This exam is designed to take 100 minutes, but we will use the full 110 minutes from 6:00-7:50PM; for 50% extra time, the expected time is 150 minutes, i.e., 5:00-7:30PM
- During the exam, **questions will not be answered** except when there is a glaring mistake or ambiguity in the statement of a question; we cannot clarify a question for you; please do your best to interpret and answer each question clearly and concisely
- Long answers are difficult to grade; the space provided should be sufficient for each question; however, you may use the last page of this exam for overflow work
- All work on this exam must be your own; do not even think of copying from others
- When you hand in your exam, be prepared to show your RPI ID

Please sign below to indicate that you will not copy or cheat on this exam:

Signature: ______________________________

Do not start this exam until you are instructed to do so.

1. **(3 POINTS)** Given undirected graph $G = (V, E)$ represented by an adjacency matrix, what is the runtime of DFS to determine whether node $t \in V$ is reachable from node $s \in V$? Assume individual lookups in the adjacency matrix are $O(1)$. Clearly circle the **best** answer.

   SOLUTION: $O(|V|^2)$

2. **(3 POINTS)** How many connected components are there in the undirected graph below? Clearly circle the **best** answer.

   SOLUTION: 3 (MAKEUP is 2)
   (i.e., $\{A, B, E, I, J\}$, $\{F\}$, and $\{C, D, G, H, K, L\}$)

3. **(3 POINTS)** Applying Dijkstra's algorithm to the directed graph below, what is the shortest distance (i.e., minimum sum of all edge weights) from node A to node F? Clearly circle the **best** answer.

   SOLUTION: 6 (5 edges MAKEUP) (i.e., path $A \to B \to C \to D \to G \to F$)

4. **(3 POINTS)** How many strongly connected components are there in the directed graph below? Clearly circle the **best** answer.

   SOLUTION: 3 (i.e., $\{A, B, E\}$, $\{C\}$, and $\{D, F, G, H, I\}$)

5. **(3 POINTS)** What is the minimum number of edges you must add to the directed graph below to make it strongly connected? Clearly circle the **best** answer.

   SOLUTION: 2 (i.e., there are five SCCs: source SCCs $\{B\}$ and $\{E\}$; SCCs $\{A\}$ and $\{G, H, I\}$; and sink SCC $\{C, D, F, J\}$; therefore, we must add an edge from any node of the sink SCC to any node of *each* source SCC, for a total of 2 such edges)

6. **(12 POINTS)** Draw an undirected graph G with five nodes and seven edges such that the `pre` and `post` numbers from the DFS algorithm for all but one of the nodes differ by at least 3 (i.e., for each node u in G, $\texttt{post}(u) > \texttt{pre}(u) + 2$).

   SOLUTION:

   - **(2 points)** graph is undirected
   - **(2 points)** graph has five nodes
   - **(2 points)** graph has seven edges
   - **(6 points)** graph has at least one node from which DFS meets the `pre`/`post` number requirements

7. **(12 POINTS)** Draw a directed acyclic graph (DAG) with four nodes that has two sources and four distinct topological orderings.

   SOLUTION:

   - **(2 points)** graph is a DAG
   - **(2 points)** graph has four nodes
   - **(4 points)** graph has two sources
   - **(4 points)** graph has four distinct topological orderings

8. **(12 POINTS)** Draw a graph with four nodes for which Dijkstra's algorithm fails to find the shortest path between a source node S and at least one other node, but the Bellman-Ford algorithm succeeds.

SOLUTION:

- **(1 points)** graph has source node S shown
- **(1 points)** graph has four nodes
- **(4 points)** Dijkstra's algorithm fails from node S to at least one other node specifically due to a negative weight that causes distance d to propagate incorrectly (see example in sample Exam 1 prep solutions)
- **(6 points)** The Bellman-Ford algorithm successfully finds shortest paths from node S to all other nodes (watch for negative cycles, which also "breaks" the Bellman-Ford algorithm)

9. **(12 POINTS)** Draw a connected undirected graph with six nodes and at least six edges in which the shortest (i.e., minimum weight) path between two nodes u and v is **not** part of any minimum spanning tree (MST). Show the shortest path, then draw all possible MSTs.

SOLUTION:

- **(2 points)** graph has six nodes
- **(2 points)** graph has at least six edges
- **(4 points)** all possible MSTs are shown
- **(4 points)** shortest path between identified nodes u and v is not part of any MST (u and v must be shown on graph)

10. **(12 POINTS)** Draw a strongly connected directed graph $G = (V, E)$ with $|V| = ??$ (varies) such that, for every $u \in V$, removing u from G leaves a directed graph that is no longer strongly connected.

SOLUTION:

- **(1 points)** graph is a directed graph
- **(2 points)** graph has specified number of nodes (varies with version of exam)
- **(3 points)** graph is strongly connected
- **(6 points)** graph is a cycle (i.e, removing any node u causes resulting graph to no longer be strongly connected)

11. **(12 POINTS)** Write an algorithm to find a path that traverses all edges of directed graph G exactly once or determines that such a path does not exist for G. You may visit nodes multiple times, if necessary. Show the runtime complexity of your algorithm.

SOLUTION:

- **(5 points)** algorithm is correct (look for counter-examples)
- **(1 points)** algorithm identifies/returns a path...
- **(2 points)** ...or determines that such a path does not exist
- **(4 points)** runtime complexity is shown and is accurate

12. **(13 POINTS)** Consider the following pseudocode of a function that takes integer $n \geq 0$ as input.

```
function netflix(n):
  print '*'
  if n == 0: return
  for i = 0 to n - 1:
    print '*'
  netflix(n - 1)
  return
```

Let $T(n)$ be the number of times the above function prints a star (`'*'`) when called with valid input $n \geq 0$. What is $T(n)$ exactly, in terms of n only (i.e., remove any reference of $T()$ on the right-hand side). Prove your statement.

SOLUTION:

- **(7 points)** $T(n)$ can be expressed as

  $$T(n) = \sum_{i=1}^{n+1} i$$

  or just

  $$T(n) = \frac{(n+1)(n+2)}{2}$$

  Note that i could start at 0 in the summation; other rearrangements of this are fine as long as $T()$ does not appear on the RHS
- **(6 points)** Proof by induction is the best approach here, though (similar to the homework) stating that this is proven by definition is also fine