COMP 282 - Midterm 1 (Spring, 2019) Name:

**Question 1 (10 Points)** Provide a short answer to the following questions. a Give an example of when you might use a linked list.

1. Give an example of when you would use a graph.
2. What is the property that allows Binary Trees to be quickly searchable? What is an “easy” way of maintaining this property?
3. We know that all trees are graphs. Why are not all graphs trees?
4. Why do we have asyptotic analysis (Big-O notation)?

**Question 2 (10 Points)** Circle the appropriate choices for the following questions. There may be more than one correct answer.

1. Which data structures are searchable in *O*(lg *n*) time. Assume any favorable arrangements of data neces- sary to achieve this property.
   1. Trees
   2. Linked Lists iii Arrays
2. Graphs
3. Binary Trees vi None of These
4. Of these, which is the most appropriate data structure with which to construct a FIFO queue? i Trees
5. Arrays
6. Primitive
7. Binary Trees v Linked Lists
8. Which of the following properties are relevant to binary trees? i Balance

ii Mass Density iii Edge Weight iv Shortest Path v Height

vi Half-Life

1. Circle both the *minimum* and *maximum* height a binary tree containing 31 values may have. i 0
2. 31
3. 6
4. 42
5. 5
6. 4

**Question 3 (10 Points)** Given the following scenarios, describe a data structure that would be most- appropriate. Justify each.

1. You are building an e-commerce site which is expected to have a lot daily visitors. The problem is, your payment system is notoriously slow to process individual payments. You want a way to keep track of all customers currently waiting to check out their orders.
2. You are writing a program for the campus library. The task is to organize all the Computer Science research papers currently in the archives. Each article has a list of references at the end that refer to other articles in the collection.

**Question 4 (10 Points)** Provide all of the listed traversals for the following binary tree. Be sure to label them.

* + Pre-order traversal.
  + In-order traversal.
  + Breadth-first traversal.



10

4

17

2

6

13

20

1

5

7

16

29

25

31

30

**Question 5 (20 Points)** Use Kruskal’s Algorithm to calculate the minimum spanning forest of the following graph *tt* = (*V, E, w*). Show all steps. List all vertices in a particular spanning tree, and give its final cost.

*V* = *{v*0*, v*1*, v*2*, v*3*, v*4*, v*5*, v*6*, v*7*, v*8*, v*9*}*

E

*{v*0*, v*1*}*

*{v*1*, v*2*}*

*{v*3*, v*4*}*

*{v*4*, v*5*}*

*{v*5*, v*6*}*

*{v*6*, v*4*}*

*{v*6*, v*7*}*

*{v*7*, v*4*}*

*{v*7*, v*8*}*

*{v*8*, v*9*}*

*{v*9*, v*5*}*

w

3

2

3

1

2

2

4

3

2

1

4

**Question 6 (20 Points)** Given the following graph *tt* = (*V, E*), use any (appropriate) algorithm dis- cussed in class to list the vertices that form a connected component with *v*3. State the algorithm you are using, show all steps.

*V* = *{v*0*, v*1*, v*2*, v*3*, v*4*, v*5*, v*6*, v*7*, v*8*, v*9*}*

*E* = *{{v*0*, v*1*}, {v*1*, v*4*}, {v*3*, v*2*}, {v*2*, v*5*}, {v*5*, v*6*}, {v*5*, v*8*}, {v*6*, v*8*}, {v*6*, v*9*}, {v*7*, v*6*}, {v*8*, v*7*}, {v*7*, v*9*}, {v*9*, v*8*}}*

**Question 7 (20 Points)** Given the graph *tt* = (*V, E, w*), below, find the shortest path between *v*2 and

*v*6. Show all steps.

*V* = *{v*0*, v*1*, v*2*, v*3*, v*4*, v*5*, v*6*, v*7*, v*8*, v*9*}*

E

*{v*0*, v*4*}*

*{v*0*, v*1*}*

*{v*0*, v*6*}*

*{v*1*, v*2*}*

*{v*1*, v*0*}*

*{v*2*, v*9*}*

*{v*2*, v*8*}*

*{v*2*, v*7*}*

*{v*3*, v*2*}*

*{v*5*, v*2*}*

*{v*5*, v*6*}*

*{v*6*, v*2*}*

*{v*6*, v*7*}*

w

1

7

2

4

7

3

6

4

1

3

4

10

2