Name:

CS 146

Final

Question 1: \_\_\_ / 10

Question 2: \_\_\_ / 10

Question 3: \_\_\_ / 10

Question 4: \_\_\_ / 10

Question 5: \_\_\_ / 10

Question 6: \_\_\_ / 10

Question 7: \_\_\_ / 10

Question 8: \_\_\_ / 20

Question 9: \_\_\_ / 5

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

100

Question 1: Big-Oh

Each bound should be as tight and simple as possible in Big-Oh in terms of the variable n.

What is the run-time for the following code?

a) (worth 3 points)

for(i = 1; i < n; i++) {

x = n;

while (x > 0) {

sum++;

x = x / 2;

}

}

b) (worth 2 points)

for (int j = 4; j < n; j=j+2) {

val = 0;

for (int i = 0; i < j; ++i) {

val = val + i \* j;

for (int k = 0; k < n; ++k){

val++;

}

}

}

c) (worth 3 points)

for(i = 0; i < n; i++) {

for(j = 0; j < i; j++) {

sum++;

}

for(k = 0; k < i; k++) {

sum++;

}

}

d) (worth 2 points)

x = n;

while (x > 0) {

y = n;

while (y > 0) {

sum++;

y = y / 2;

}

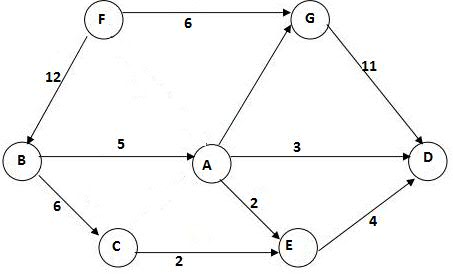
x = x / 2;

}

Question 2: Sorting (worth 10 points)

You have a 20 GB file with one string per line. Explain how you would sort the file?

Question 3: Graph Basics & Representation (worth 10 points)



a) Draw an adjacency matrix representation of this graph. (worth 2 points)

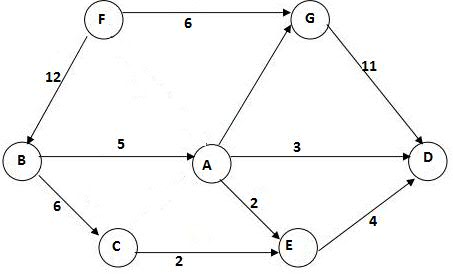
b) Draw an adjacency list representation of this graph. (worth 2 points)

1. Provide an O (Big-Oh) bound on the space used by the adjacency matrix representation. (worth 2 points)
2. Provide an O (Big-Oh) bound on the time to check whether two vertices are adjacent using the adjacency list representation. (worth 2 points)

e) Which of these two representations was emphasized as being much more commonly

used in representing real world problems, and why? (worth 2 points)

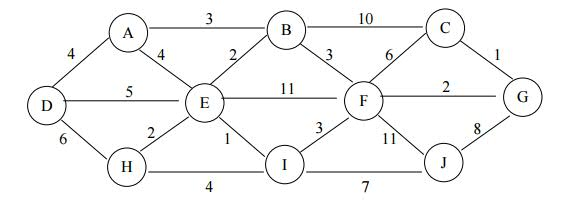
Question 4: Topological Sort (worth 10 points)



a) Starting from node F, what is a topological sort? (worth 6 points)

b) Is the sort you found unique? Why? (worth 4 points)

Question 5: Kruskal's (worth 10 points)



Apply Kruskal’s algorithm to compute a minimum spanning tree. In the designated

spaces below, write down the edges in the order they are considered by Kruskal’s

algorithm. If the edge is part of the minimum spanning tree found by the algorithm, write

it down in the first list of edges that form the MST. Write down the other edges

considered by the algorithm in the order they were considered. Assume that the algorithm

terminates as soon as the MST has been found.

In the lists, use (x,y) to indicate an edge connecting vertices x and y.

a) Edges that form part of the MST, in order considered:

b) Other edges considered, but not included in the MST, in order considered:

c) Is the MST you found unique? Why?

Question 6: P = NP? (worth 10 points)

a) What is the question of P = NP? Describe under three sentences. (worth 5 points)

b) Is Towers of Hanoi in NP-Complete? Explain in under three sentences. (worth 5 points)

Question 7: Fork Join (worth 10 points)

a) Given a program where 80% of it is parallelizable (and 20% of it must be run sequentially)

what is the maximum speedup you would expect to get with 5 processors. Note: You must

show your work for any credit. For full credit give your answer as a number or a

simplified fraction (not a formula). (worth 5 points)

b) What is the difference between parallelism and concurrency? Explain in under three sentences. (worth 5 points)

Question 8: Algorithm (worth 20 points)

Implement an algorithm that takes an adjacency list and produces a topological sort of the vertices. Assume no cycles. Please use pseudocode. You can use Java if you'd like.

Question 9: Extra Credit (worth 5 points)

Write a program to swap odd and even bits in an integer with as few instructions as possible (e.g., bit 0 and bit 1 are swapped, bit 2 and bit 3 are swapped, and so on).