COS 226 **Algorithms and Data Structures** **Spring 2019**

Final

This exam has **10 questions** worth a total of **118 points**. You have **180 minutes**. The exam is preprocessed by a computer, so please write inside the designated spaces.

The exam is closed book, except that you are allowed to use a **one page sheet of notes** (8.5-by-11 paper, both sides, in your own handwriting). **No electronic devices** are permitted.

Do not discuss the contents of this exam with students who have not taken it yet. Do not remove this exam from this room.

Name:

NetID:

Exam Room:

Precept:

P01	P01A	P02	P03	P04	P04A	P05	P05A	P05B	P06
○	○	○	○	○	○	○	○	○	○

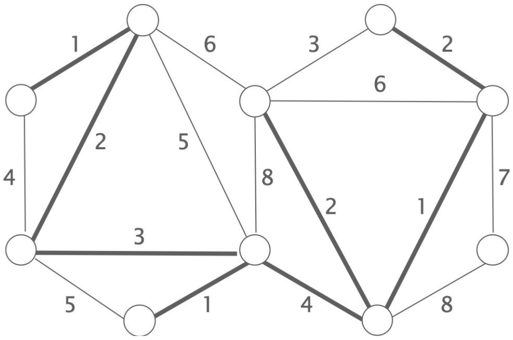
Write this in the box below:

"I pledge my honor that I will not violate the Honor Code during this exam."

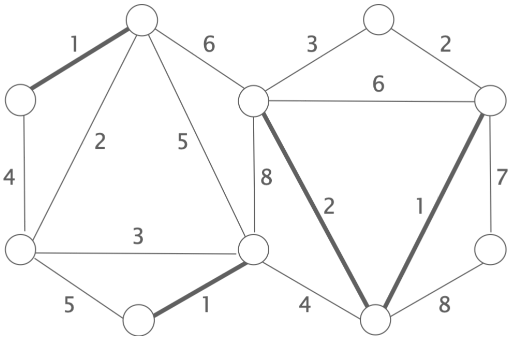
Signature

1. Minimal spanning tree [12 points]

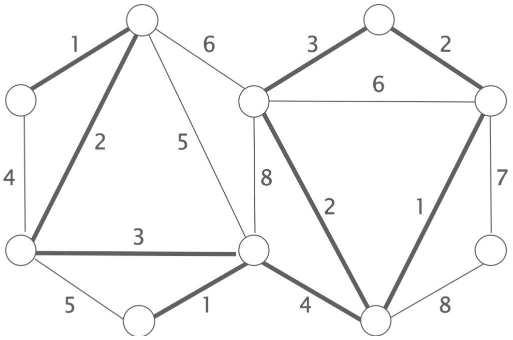
A set of edges is highlighted in different copies of the same graph in each of the figures below. Determine whether it could possibly represent a partial spanning tree obtained from (a prematurely stopped) Prim's algorithm, (a prematurely stopped) Kruskal's algorithm, both, or neither. Write "Prim's", "Kruskal's", "Both", or "Neither" the blank below each figure.



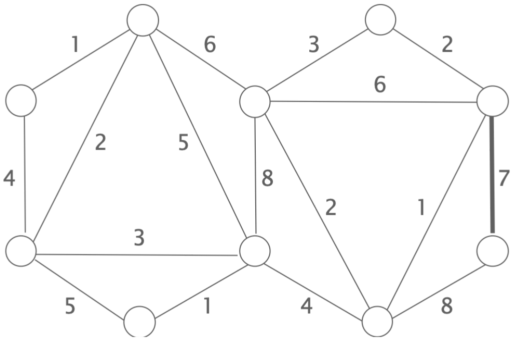
____



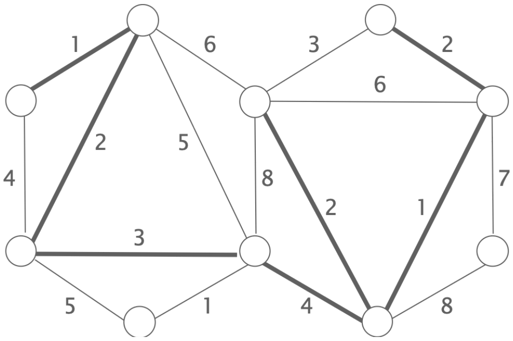
____



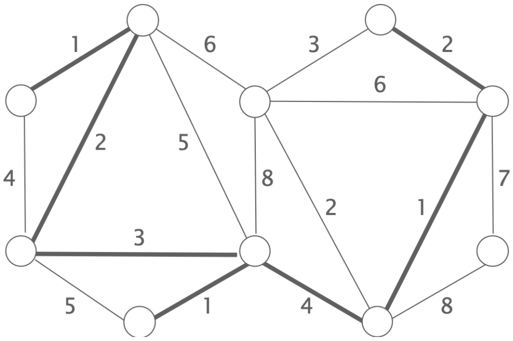
____



____



____



____

2. 2d tree [8 points]

Figure 1

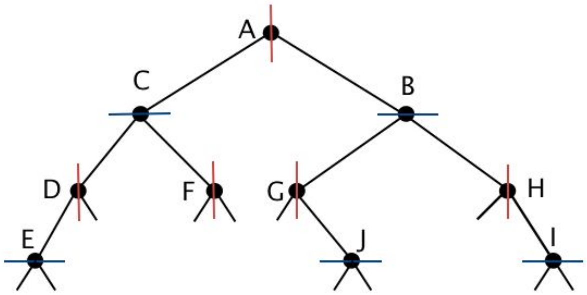
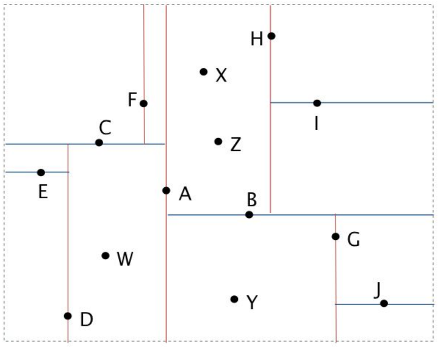


Figure 2



A 2d-tree contains ten points (denoted A, B, ... J). The tree is shown in Figure 1 and the points are shown in Figure 2.

A. Points W, X, Y, and Z with the coordinates shown in Figure 2 are inserted into the tree in that order. Show where the four new points will be inserted by filling in the blanks below. (Write 'left' or 'right' in the first blank of each statement).

W will be the __________ child of ____.

X will be the __________ child of ____.

Y will be the __________ child of ____.

Z will be the __________ child of ____.

B. Suppose we do a nearest neighbor search for a point with the same x- and y-coordinates as D (before inserting W, X, Y, Z). How many distance computations between the query point and a point in the tree will be performed?

______

C. Complete the following sentence with a succinct phrase. In general, if we do a nearest-neighbor search for a query point that equals a point *P* that's already in a kd-tree, the number of distance computations between points that will be performed is

____.

3. Boyer-Moore [9 points]

A. Suppose we search for the pattern `WAYNE` in the text `BOBSEDGEWICKWEYNEKEVIN`.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
B	O	B	S	E	D	G	E	W	I	C	K	W	E	Y	N	E	K	E	V	I	N

W	A	Y	N	E
0	1	2	3	4

Show the sequence of pairs of indices that will be compared by the Boyer-Moore algorithm by filling in the blanks below. The first pair is already shown. Note that there may be more blanks than necessary.

Index *i* in text: 4 ___ ___ ___ ___ ___ ___ ___ ___ ___

Index *j* in pattern: 4 ___ ___ ___ ___ ___ ___ ___ ___ ___

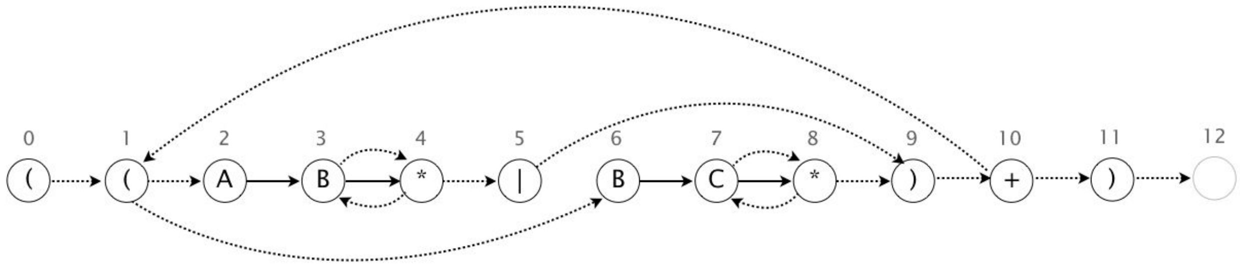
B. Suppose we search for the pattern `AA...A` of length *m* in the text `BAA..ABAA..ABAA..A`, consisting of repeated chunks of a single '`B`' followed by *m*-1 '`A`'s. For example, if *m*=3, then the pattern is `AAA` and the text is `BAABAABAA...`

How many character comparisons will the Boyer-Moore algorithm make? Express your answer in tilde notation as a function of the length of the text *n* and the length of the pattern *m*. You may assume that *n* is much larger than *m*.

~ □

4. NFA [10 points]

Consider the following NFA:



A. Write the regular expression that this NFA matches:

____

B. Select *all* states that the NFA can be in after processing the string ABB.
If the NFA gets stuck while processing the string, select *None*.

None	0	1	2	3	4	5	6	7	8	9	10	11	12
□	□	□	□	□	□	□	□	□	□	□	□	□	□

C. Select *all* states that the NFA can be in after processing the string AACBA.
If the NFA gets stuck while processing the string, select *None*.

None	0	1	2	3	4	5	6	7	8	9	10	11	12
□	□	□	□	□	□	□	□	□	□	□	□	□	□

D. Does the NFA accept the string BBB? Write Yes or No.

□

E. Does the NFA accept the string ABBBBCCCC? Write Yes or No.

□

5. Compression [12 points]

A. Decode the following LZW-encoded bitstream represented as a sequence of hex characters:

46 41 44 45 44 81 43 82 84 45 86 84 80

Decoded string: ____

Recall that:

1. The ASCII values of the uppercase letters in hex are A: 41, B: 42, C: 43, etc.
2. We use hex 80 as a stop symbol, and codes for new codewords start at hex 81.

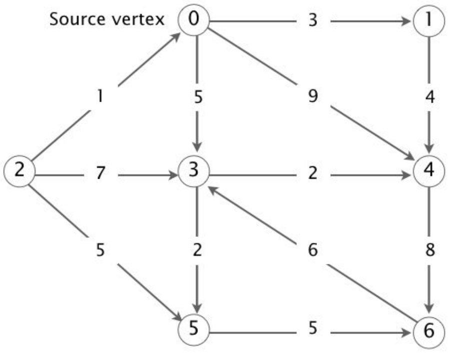
Show the codeword table that results from decoding by filling in the cells below. Note that there may be more cells in the table than necessary.

Code	81	82	83	84	85	86	87	88	89	8a	8b
Codeword	____	____	____	____	____	____	____	____	____	____	____

B. Construct a Huffman code for the string "`SLEEPLESSNESS_LESSENS_SENSE`". Count the frequency of each of the six characters (including the underscore) that appear in the string, and write the binary code for each.

	Freq	Code
E	____	________
L	____	________
N	____	________
P	____	________
S	____	________
_	____	________

6. Shortest paths [12 points]



A. Simulate Dijkstra's algorithm on the above edge-weighted digraph, starting from vertex 0. Show the resulting `distTo[]` and `edgeTo[]` arrays below. Recall that `distTo[]` entries are initialized to ∞ and `edgeTo[]` entries are initialized to `null`.

Vertex	0	1	2	3	4	5	6
`distTo`	____	____	____	____	____	____	____
`edgeTo`	____	____	____	____	____	____	____

B. For each of the sentences below, write 'T' or 'F' in the corresponding box to indicate whether the statement is true or false. Note that when the statements refer to a digraph, it is not necessarily the digraph in part 'A' above. Assume positive edge weights throughout.

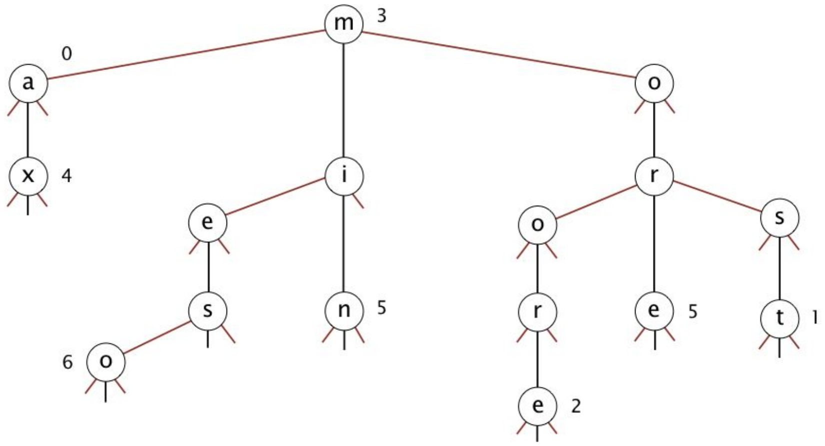
If we run different implementations of Dijkstra's algorithm on the same digraph, the values of the resulting `distTo[]` array are guaranteed to be the same. □

If we run different implementations of Dijkstra's algorithm on the same digraph, the values of the resulting `edgeTo[]` array are guaranteed to be the same. □

If we run different implementations of Dijkstra's algorithm on the same digraph with distinct edge weights, the values of the resulting `edgeTo[]` array are guaranteed to be the same. □

Suppose different implementations of Dijkstra's algorithm use the same priority queue code that breaks ties in `delMin()` based on the order of insertion (i.e. the least recently added minimum element will be returned first). If we run those implementations on the same digraph, then the values of the `edgeTo[]` array are guaranteed to be the same. □

7. String symbol tables [11 points]



A. A Ternary Search Trie (TST) is shown above. Values are indicated with a number next to the node. List all the strings that the TST contains. Note that there might be more spaces below than necessary, and that the strings might not be English words. There is a score penalty for incorrect answers.

________   ________   ________   ________   ________   ________

________   ________   ________   ________   ________   ________

B. For each of the following three string symbol table implementations, indicate how characters and strings are stored by picking the best match from among three choices:

	Linear probing Hash table	R-way trie	TST
Neither characters nor strings are stored explicitly	○	○	○
Characters are stored explicitly but strings are not	○	○	○
Strings (and characters) are stored explicitly	○	○	○

C. Suppose a million random strings, each of length 10, over a 256-character alphabet are inserted into a symbol table.

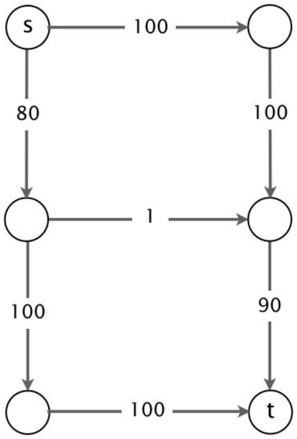
Among the above three implementations, which one would consume the most space?

Answer: □

Which of the three would consume the least space?

Answer: □

8. Max-flow [10 points]



A. In the flow network above, what is the worst-case number of augmenting paths found by the Ford Fulkerson algorithm if augmenting paths are selected in an arbitrary way (i.e., no heuristic is used)?

Answer:

B. What is the worst-case number of augmenting paths if the shortest path heuristic is used to select augmenting paths, with ties broken arbitrarily?

Answer:

C. What is the worst-case number of augmenting paths if the fattest path heuristic is used to select augmenting paths, with ties broken arbitrarily? Recall that this heuristic selects the path with the maximum bottleneck capacity.

Answer:

9. True or False [14 points]

For each statement below, write ‘T’ or ‘F’ in the corresponding box to indicate whether the statement is true or false.

Suppose we explore the right/top subtree before the left/bottom subtree in 2d-tree range search. Then it may return the wrong answer. □

Suppose we explore the right/top subtree before the left/bottom subtree in 2d-tree range search. Then it may explore more nodes. □

Ignoring the encoding of the trie, the Huffman encoding of a message will never be longer than the message itself (assume that the message is represented using a fixed-length code such as ASCII). □

Let G be a connected edge-weighted graph and G' be a graph obtained by adding a single edge to G. Then G has an MST T and G' an MST T' such that there is at most one edge in T that is not in T'. □

For any regular expression with m characters, there is a DFA with at most m^2 states that recognizes the same set of strings matched by the regular expression. □

Given a flow network with V vertices and E edges, as well as a min-cut of the network, the value of the max-flow can be computed in time proportional to $E+V$. □

MSD radix sort is applicable only when the input strings all have the same length. □

This page is intentionally left (nearly) blank. Feel free to use it as scratch space. Exam continues on the following page.

10. Data structure and algorithm design [20 points]

By the late 19th century, most of the world's major cities were connected by the telegraph network. Your goal in this problem is to model this network to understand how quickly news would propagate around the world.

Each telegraph link connects a pair of cities. When news originates in a city, it is sent to each of the cities to which it is linked. The first time a city hears a piece of news, it will send that news to each of the cities to which it is linked.

Some links are congested, which means that messages will have to wait exactly one hour before being sent through them (in either direction). There are no other delays in the system: when a city hears a piece of news, all cities linked to it via uncongested links immediately hear it as well. The network is given to you as a sequence of pairs of cities, along with congestion status:

```
London       New_York       Congested
New_York     Atlanta        Uncongested
Washington   Philadelphia   Congested
…
```

There are *V* cities and *E* telegraph links in the input. Assume that the lengths of city names are at most some constant number.

Your answers below will be graded for correctness, efficiency, and clarity. Answer in prose, but feel free to use some pseudocode if you think it will improve clarity. You may make any standard technical assumptions that we have seen in this course.

A. Describe how you will pre-process the input so that you can quickly answer queries about propagation time for news originating in any given city (i.e. the queries in parts C-E). For full credit, this preprocessing must complete in time proportional to *E* log *V* (or better) in the worst case.

Hints:

1. Read the entire question before answering part A, so that you understand what is required of the preprocessing step.
2. In our reference solution, the majority of the text is in part A, whereas the solutions to parts C-E are relatively succinct.

B. What is the order-of-growth running time of your solution to part A? Briefly justify your answer.

C. Given the names of two cities, determine whether they are connected via a path of telegraph links. For full credit, this must complete in time proportional to log V (or better) in the worst case.

D. Given the names of two cities, determine whether news originating in one of them will reach the other instantly (i.e., via uncongested links). For full credit, this must complete in time proportional to log V (or better) in the worst case.

E. Given an originating city, sort the V cities based on the time at which they will hear news originating in the given city. If two cities hear about the news at the same time, they may be sorted relative to each other in either order. For full credit, your algorithm must complete in time proportional to $E + V$ (or better) in the worst case.

F. What is the order-of-growth running time of your solution to part E? Briefly justify your answer.