School of Computing and Information Technology

Student to complete:

Family name	Abdullah
Other names	Kazi Swad
Student number	5220683

CSCI203/CSCI803
Algorithms and Data Structures
Wollongong Campus

Examination Paper
Spring Session 2020

Exam duration	3 hours
Weighting	50%
Items permitted by examiner	Lecture notes
Aids supplied	None
Directions to students	All questions to be answered. The value of each question is printed beside each question. .

Question 1: **8 marks**

a. Explain how the A* heuristic can be used to accelerate the process of finding the shortest path between two specified points in a graph.

We will take the minimum of the actual distance between the two points and the euclidean distance. The closer the euclidean distance is to the actual distance the faster we will find the solution.

b. What data, in addition to the weighted graph, are required for A* to be used?

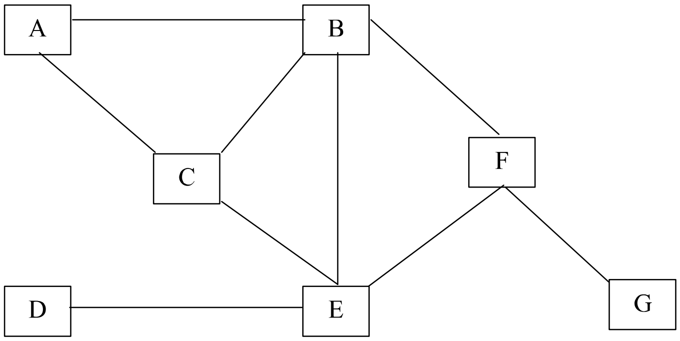
In addition the the weighted graph we need the x and y coordinates of the two points in the graph. This will be used to find the euclidean distance between the two points

c. What conditions must the problem meet before A* can be used?

The euclidean distance must be less than the the weighted distance. H(v,g) <= P(v,g).

Question 2: **8 marks**

Consider the following graph:



a. Using a breadth first search and selecting nodes in alphabetical order, list the edges that form the spanning tree for this graph, in the order that they are found by the search.

A-C
A-B
C-E
B-F
E-D
F-G

b. Using the above graph as an example, explain how you would find its articulation points.

A vertex v, of a connected graph is an articulation point if the graph obtained by deleting node v and all its edges is no longer connected.

A simple approach to finding the articulation points is to one by one remove all vertices and see if removal of a vertex causes disconnected graph

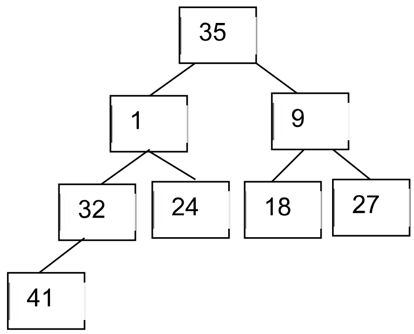
the algorithm to find the articulation point is shown below:

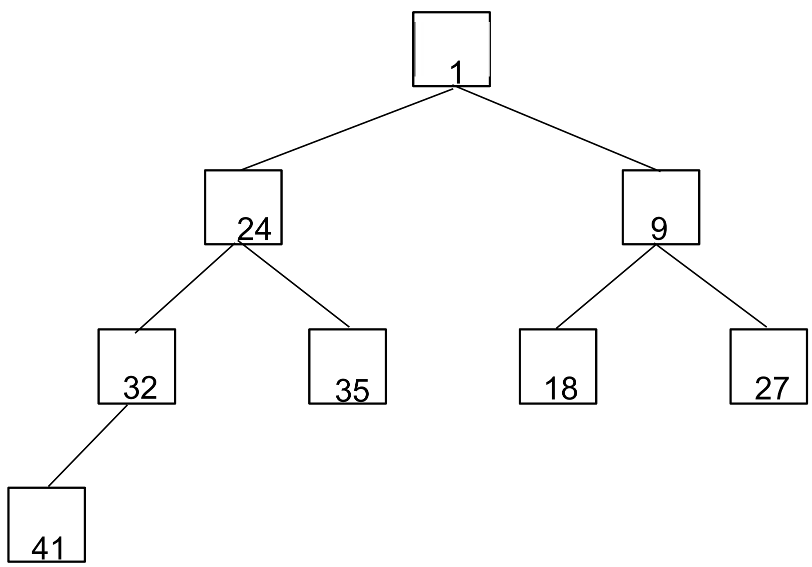
* Remove vertex from graph
* See if the graph remains connected.
* Add the vertex back to the graph

In doing so we find that nodes F and G are the articulation points of the graph above.

Question 3: **8 marks**

a. Show the tree resulting from the operation of the siftdown function, which is called on the root node to restore the min heap property for the following tree:





b. What is the complexity of the siftdown function? Explain.

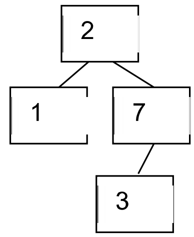
Complexity of the siftDown function is O(logN).

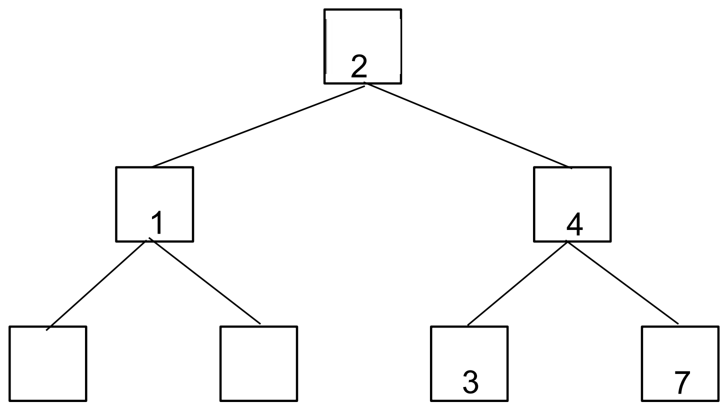
We recursively call siftdown with half the dumber of elements n/2 after two comparisons.

Therefore, T(n) = 2 + T(n/2).
T(n) = 2logN.

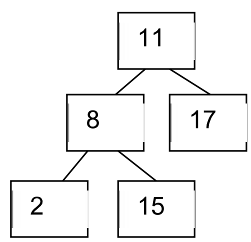
Thefore, O(n) = logN.

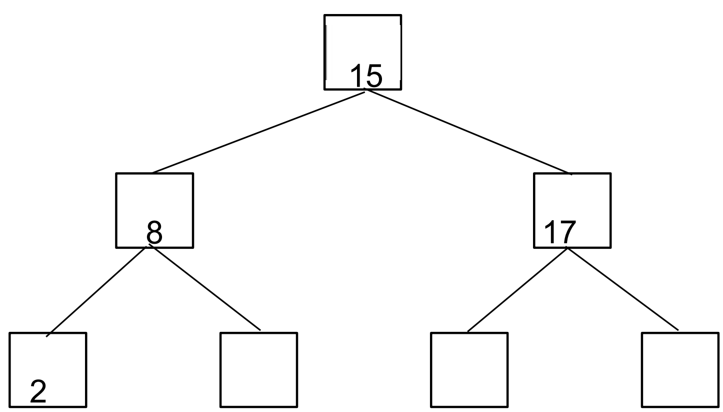
c. Draw the AVL tree resulting from the insertion of the node 4, into the tree:





d. Draw the AVL tree resulting from the deletion of the node 11 from the tree:





Question 4: **8 marks**

Describe an algorithm to efficiently find a peak, a local maximum, in:

a. A one-dimensional array of positive integers.

- We find the middle value of the remaining array of integers.
- If the first value to the left of the middle value is greater than the middle value, then we reduce the array by half and set the end point of the array to the first value to the left of the middle value.
- If the first value to the right of the middle value is greater than the middle value, then we reduce the array by half and set the start point of the array to the first value to the right of the middle value.
- We repeat the above process until we find that either the first value either to the right and left of the middle value is less than or equal to the middle value, or we have reached the start or end of the array.

b. If the array contains *N* values, what is the complexity of the above peak-finding algorithm?

If an array contains N values, the complexity is O(logN)

c. A two-dimensional array of positive integers.

- We find the largest element in the middle column of the remaining array of integers.
- If the element to its left is larger than the maximum, we discard the right half of the array.
- If the element to its right is larger than the maximum, we discard the left half of the array.
- We repeat the above process until we find that the element to its right or left is smaller then or equal the maximum, then the maximum is the peak.

d. If the array contains M rows of *N* values, what is the complexity of the above peak-finding algorithm?

If an array contains M and N values, the complexity is O(MlogN)

Question 5: **9 marks**

Briefly describe how the following operations are performed in a 2-4 tree.

a. Insertion into a node of the tree which currently has 3 occupants. Explain both possible actions and under what circumstances each is taken.

b. Deletion from a node which currently has a single occupant. Explain both possible actions and under what circumstances each is taken.

Question 6: **9 marks**

Dynamic Programming can be used to develop an algorithm using the 5 steps shown in lectures.

Using the computation of the Fibonacci numbers as an example, explain the process by which an efficient algorithm to compute Fib(n) can be created using the steps of the DP process.