The University of Nottingham

SCHOOL OF COMPUTER SCIENCE

A LEVEL 2 MODULE, AUTUMN SEMESTER 2015-2016

ALGORITHMS AND DATA STRUCTURES

Time allowed NINETY Minutes

Candidates may complete the front cover of their answer book and sign their desk card but must NOT write anything else until the start of the examination period is announced

Answer ALL FOUR questions

Marks available for sections of questions are shown in brackets in the right-hand margin

No calculators are permitted in this examination.

Dictionaries are not allowed with one exception. Those whose first language is not English may use a standard translation dictionary to translate between that language and English provided that neither language is the subject of this examination. Subject specific translation dictionaries are not permitted.

No electronic devices capable of storing and retrieving text, including electronic dictionaries, may be used.

DO NOT turn your examination paper over until instructed to do so

ADDITIONAL MATERIAL: none

INFORMATION FOR INVIGILATORS: Students should write all their answers in the answer book. The exam paper should be collected and placed inside the answer book.

1. This compulsory question concerns a variety of topics. (25 marks total)

(a) Give appropriate descriptions in terms of each of \`O′ (big-Oh) and \`Ω′ (big-Omega) of the function

$$f(n) = 2\,n^3 + n^2 \log(\,n^5\,)$$

You should justify your answers, but do not need to give proofs.

(5 marks)

(b) Consider the following code implementing selection sort of an array, A, of integers into non-decreasing order (note that the array indices start at 0 as usual)

```
1.   void sort( int[] A )
2.   {
3.        for ( int i = A.length – 1 ; i > 0 ; i-- )  {
4.              int p = 0;
5.              for ( int k = 1 ; k <= i ;  k++ )  {
6.                    if ( A[k]  > A[p] )  {
7.                          p = k;
8.                    }
9.              }
10.             if (  i != p ) {
11.                   int temp = A[p];
12.                   A[p] = A[i];
13.                   A[i] = temp;
14.             }
15.             assertTrue( S );
16.       }
17.   }
```

Give an appropriate and useful choice for assertion S on line 15, and justify your answer. You may write your answer using either a logical expression or as pseudo-code.

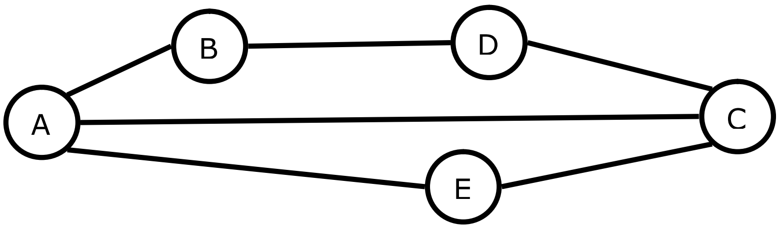
Also, state whether the above version of selection sort is stable, and justify your answer.

(8 marks)

(c) Consider an arbitrary binary tree containing n nodes. Give the big-Oh of the worst case of the height of the tree and the big-Omega of the best case of the height of the tree; and give a brief justification of your answers.

(5 marks)

(d) Consider the following graph



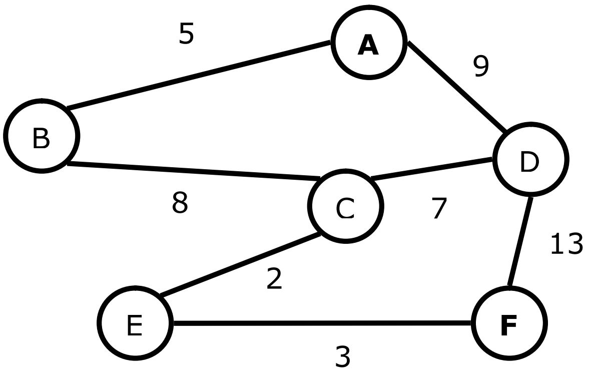
List all possible orders in which nodes can be visited during a **depth-first** traversal starting from node D. For each possible traversal you only need to give the sequence of nodes, writing it in the format [D _ _ _ _].

Include a brief explanation of your answers.

(7 marks)

2. This compulsory question concerns graphs. (20 marks total)

(a) Explain, and give pseudo-code for, Dijkstra's algorithm to find the shortest path in an undirected graph when the distances for edges are given and all are strictly positive (greater than zero). Use Dijkstra's algorithm to find the shortest path from node A to node F in the graph below. Show your working, including the open and closed lists at each stage.



(8 marks)

(b) State, and give a brief justification of, the run-time complexity of Dijkstra's algorithm as expressed in terms of the number of vertices, V, and number of edges, E, of the graph.

(3 marks)

(c) Define and explain the concept of a "Minimum Spanning Tree" (MST) in an undirected weighted graph. Then give an algorithm to find a MST, and show it working on the graph in part (a).

(6 marks)

(d) Suppose that in some graph G, there are two nodes A and B, for which the shortest path, P, between A and B passes through all the other nodes of G.

Can the length of P be smaller than the total weight (sum of lengths of the edges) of the MST of G? Justify your answer.

(3 marks)

3. This compulsory question concerns Maps and Heaps. (20 marks total)

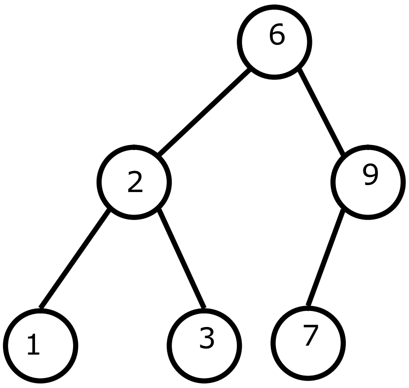
(a) State what it means for a binary tree to have the binary search tree (BST) property.

(2 marks)

(b) State what it means for a binary tree to have the heap property. Compare and contrast the heap property and the BST property.

(4 marks)

(c) The following diagram shows the initial state of a binary search tree (BST)



Show and explain your working for the resulting state of the binary search tree after performing each step of the following sequence of operations:

i) insert(8), and then

ii) delete(2), and then

iii) delete(6) (7 marks)

(d) Explain the standard method by which binary heaps can be stored using an array, with the root of the heap being mapped to index 1.

In particular, explain why the method is correct in that two different nodes will never be mapped to the same index in the array.

Also, explain the advantages of such a representation as compared to a direct representation based upon linked nodes.

(7 marks)

4. This compulsory question is concerned with the ‘big-Oh’ family. (10 marks total)

(a) Give the definitions of big-Oh, big-Omega, and big-Theta by completing each of the following:

i) ‘big-Oh’: f(n) is O(g(n)) ….

ii) ‘big-Omega’: f(n) is Ω(g(n)) ….

iii) ‘big-Theta’: f(n) is Θ(g(n)) ….

(4 marks)

(b) Given the function

$$f(n) = \begin{cases} 2\,n - 1 & \text{when n is even} \\ 3\,n + 1 & \text{when n is odd} \end{cases}$$

From the definitions prove or disprove that f(n) is Θ (n) (big-Theta)

(6 marks)