S228/212

DUBLIN INSTITUTE OF TECHNOLOGY
KEVIN STREET, DUBLIN 8

BSc (Honours) Degree in Computer Science

YEAR 2

Semester 2 Examination
2013/2014

CMPU 2001

Algorithms & Data Structures

Mr R. Lawlor
Dr D. Lillis
Mr P. Collins

Thursday 15th May
4.00 pm - 6.00 pm

Two hour exam
Attempt three out of four questions
All questions carry equal marks
One complementary mark for paper

1. (a) Provide a C# class based on a linked list implementation for the abstract data type *Queue* showing data structure and method declaration, but not method body or definition. You need only concern yourself with the two significant queue methods and the data structure. You can assume the queue stores *int* values.

(6 marks)

(b) Give the implementation of the method *enQueue()*. Explain why you might use or not use a tail reference in your implementation. What is the complexity of enQueue() with and without the use of a tail pointer?

(9 marks)

(c) Use diagrams to show how an integer can be inserted into a sorted linked list. Write code for this insert method.

(12 marks)

(d) Write down a simple equation for a Stack in terms of *pop()* and *push()* that expresses the last-in-first-out (LIFO) behaviour of the Stack.

(6 marks)

2. (a) Explain the terms:

i) priority queue
ii) complete binary tree
iii) heap
iv) heap condition

(7 marks)

(b) Draw the following heap array as a two-dimensional binary tree data structure:

k	0	1	2	3	4	5	6	7	8	9	10	11
a[k]		13	10	8	6	9	5	1				

(3 marks)

(c) Write the algorithms for the *siftUp()* and *insert()* operations on a heap. State the complexity of *siftUp()* and briefly explain why it is so.

(9 marks)

(d) By using tree and array diagrams, illustrate the effect of inserting a node whose key is **12** into the heap in the table of part (b).

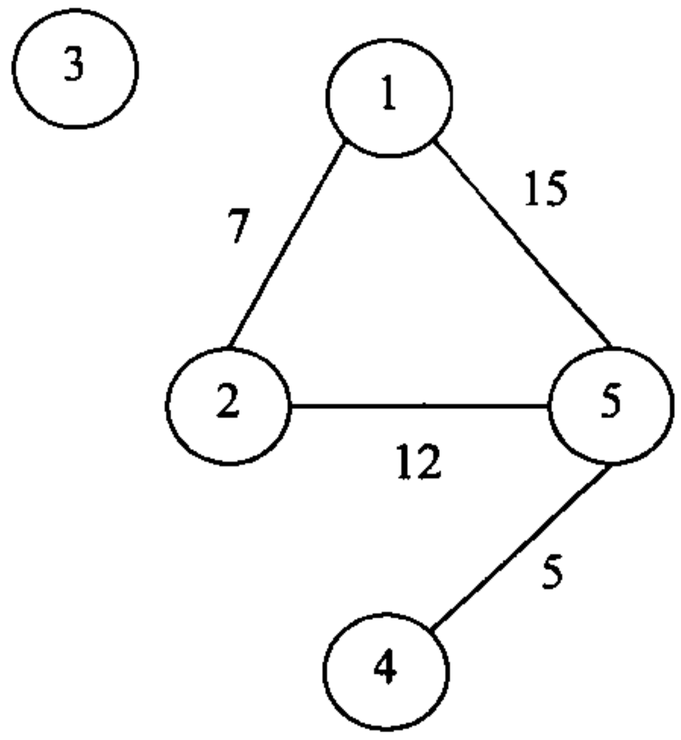
(6 marks)

(e) Given the following array, describe with the aid of text and tree diagrams how it might be converted into a heap.

k	0	1	2	3	4	5	6	7	8
b[k]		2	9	18	6	15	7	3	14

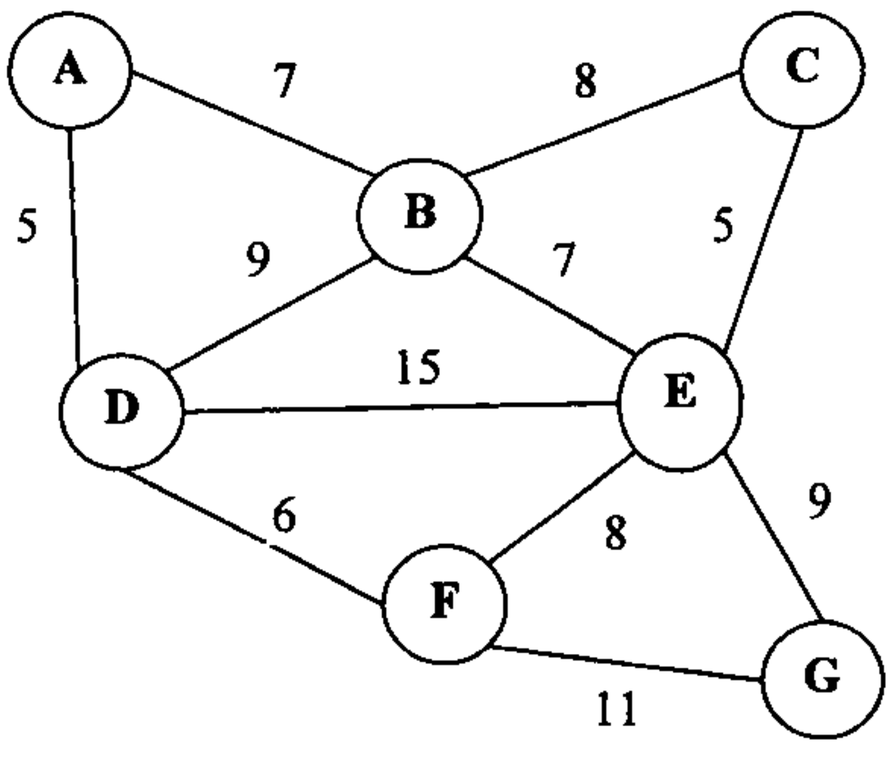
(8 marks)

3. (a) Suggest two possible representations for a weighted graph for Prim's minimum spanning tree (MST) algorithm when it is to be stored in RAM, and mention if any representation is preferable in some circumstances and why. Then illustrate how the following graph would be stored in both representations.



(7 marks)

(b) Illustrate in detail how Kruskal's algorithm computes a MST for the gra below showing the union-find sets at each stage.



(9 marks)

(c) Describe an efficient way to represent union-find sets and explain with the aid of diagrams how the union/find operations transform these sets.

(12 marks)

(d) Write down the complexity of Kruskal's algorithm and contrast it with Prim's.

(5 marks)

4. (a) Show how binary search works when searching for *17* in the following array:

1	5	6	9	10	12	14	17	21

(6 marks)

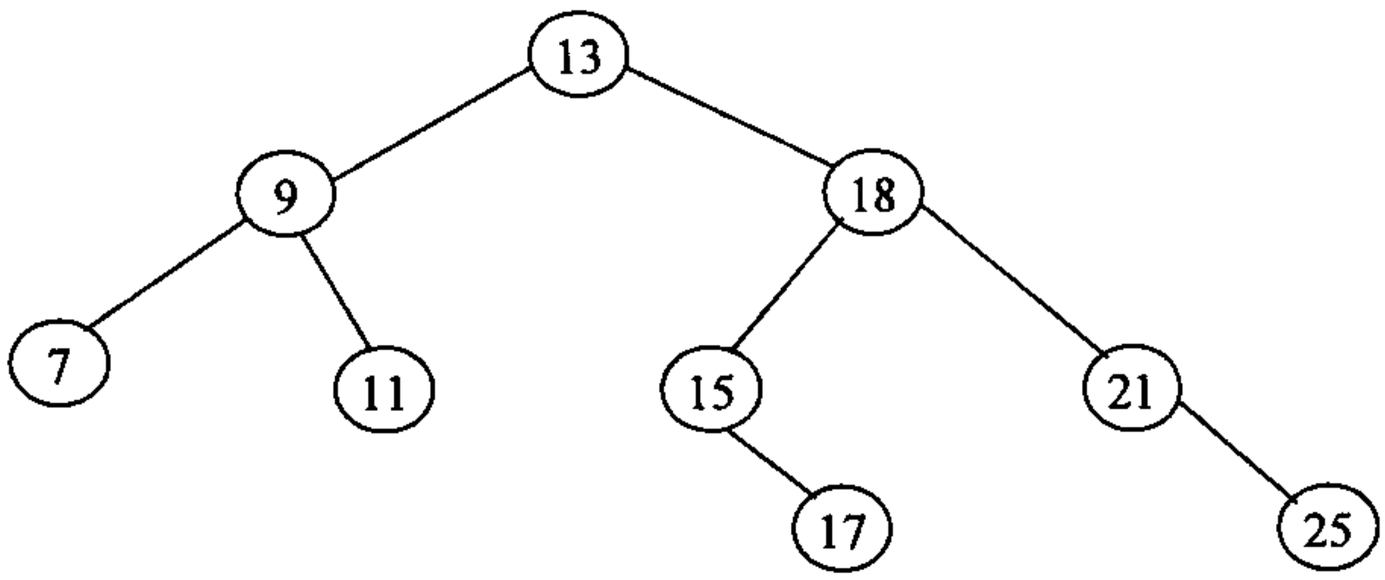
(b) Write an algorithm for binary search on a sorted array in pseudocode or code. What is its complexity?

(6 marks)

(c) What is a binary search tree? Mention any specific advantage or possible disadvantage.

(6 marks)

(d) Show the effect of inserting keys 20 and 19 into the following binary search tree, and after that the effect of removing 18.



(6 marks)

(e) How might the tree shown in part (d) be represented?
Write pseudocode or code to insert a key into a binary search tree.

(9 marks)