S228/212



DUBLIN INSTITUTE OF TECHNOLOGY



BSc. (Honours) Degree in Computer Science

Year 2



SUNIMER EXAMINATIONS 2014/2015



ALGORITHMS & DATA STRUCTURES [CMPU20011

MR. RICHARD LAWLOR

DR. DEIRDRE LILLIS

MR. PAUL COLLINS

FRIDAY 15TH MAY 4.00 P.M. — 6.00 P.M.

Two HOURS

ATIEMPT THREE OUT OF FOUR QUESTIONS

ALL QUESTIONS CARRY EQUAL MARKS

ONE COMPLEMENTARY MARK FOR PAPER

1. (a) Provide a Java or 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)

Class QueuelinkedList implements queue

{

LinkedList head, tail;

Int size;

Public QueueLinked

}

(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)

1. (a) Explain the terms:
2. priority queue
3. complete binary tree
4. heap
5. heap condition
   * + 1. m(5 marks)

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

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  | 10 | 1 1 |
|  | 13 | 10 | 8 | 6 | 9 | 5 | 1 |  |  |  |  |

k

Also, assuming another array hPos[] is used to store the position of each key in the heap, show the contents of hPos[] for this heap.

* + - 1. M(6 marks)
  1. Write in pseudocode the algorithms for the siftUp() and insert() operations on a heap and show how hPos[] would be updated in the siftUp() method if it was to be included in the heap code. Also write down the complexity of siftUp().

(9 marks)

* 1. 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). You can ignore effects on hPos[].

(6 marks)

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

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |
|  | 2 | 9 | 18 | 6 | 15 | 7 | 3 | 14 |

k

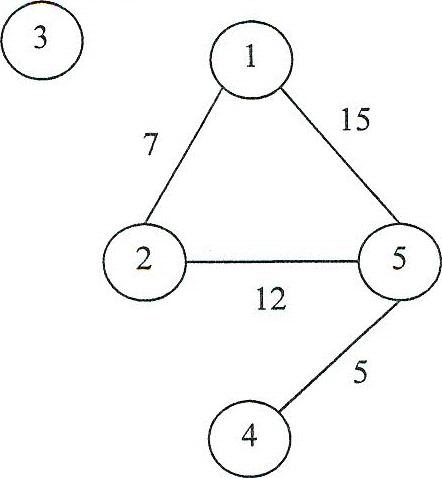
(7 marks)

1. (a) An industrial process requires the drilling of 4 holes at predefined positions in a metal sheet. The holes are bored by a robot arm which moves the drill from position to position. When the drilling is complete, the robot arm returns to its initial position. The same set of holes are to be drilled in many metal sheets, so minimising the time spent in movement of the robot arm is essential.

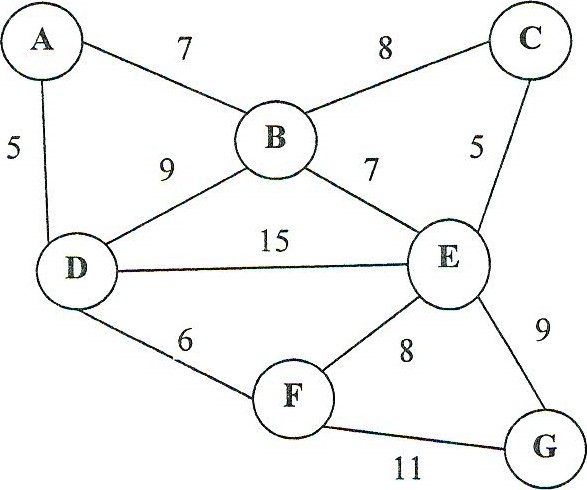
How can this problem be related to graph theory and what kind of graph if any does it correspond to?

* + - 1. marks)

(b)Suggest two possible representations for a weighted graph when it is to be stored in RAM, and mention if one representation is preferable in some circumstances and why. Then illustrate how the following graph would be stored in both representations.



* + - 1. marks)
  1. Illustrate in detail how Prim's algorithm computes a MST for the graph below showing the heap contents and the state of the arrays distance[] and parent[] at each stage. It is not necessary to show the heap tree structure at each stage, just its contents.



(10 marks)

* 1. Write Prim's algorithm in pseudocode for an adjacency lists representation of a sparse graph and discuss its complexity. What would be its complexity for a dense graph?

(12 marks)

Prim\_Lists( Vertex s )

Begin

// G = (V, E)

foreach v is an element of V

dist[v] := ̕maxValue

parent[v] := 0

// treat 0 as a special null vertex

hPos[v] := 0

// indicates that v  heap

h = new Heap(|𝑉|,ℎ𝑃𝑜𝑠,𝑑𝑖𝑠𝑡) // priority queue (heap) initially empty

h.insert(s) // s will be the root of the MST

**while** (not h.isEmpty() ) // should repeat |V|-1 times

v := h.remove() // add v to the MST

dist[v] := -dist[v] // marks v as now in the MST

***foreach*** u  adj(v) // examine each neighbour u of v

if wgt(v, u) < dist[u]

dist[u] := wgt(v, u)

parent[u] := v

**if** u is not an element of h

h.insert( u)

else

h.siftUp( hPos[u])

end **if**

end ***for***

end **while**

return parent

End

discuss its complexity: O( E log V )

dense graph: O(V squared) = O(E)

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

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 5 |  |  |  | 12 | 14 | 17 | 21 |

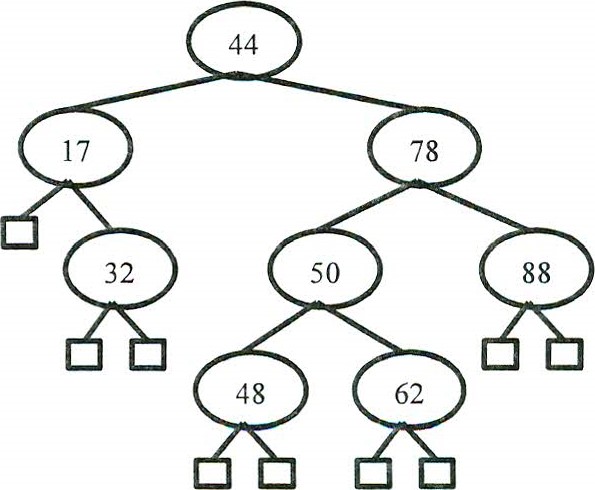
(5 marks)

(b) What is a binary search tree (BST)? Mention any specific advantage or possible disadvantage. What is the complexity of searching a BST?

(6 marks)

* 1. Write in pseudocode the algorithm for searching a BST.

(6 marks) (d) Given the following binary search tree, show how it would be modified by inserting 54.



(5 marks)

(e) What is an AVL-tree? Include in your answer the idea of a rotation.

Show how the tree that results from inserting 54 in part (d) would be rebalanced if it were an AVL-tree.

(l I marks)