Beijing-Dublin International College

SEMESTER 2 FINAL EXAMINATION - 2016/2017

School of Computer Science

COMP2003J Data Structures and Algorithms 2

Prof. Pádraig Cunningham
Dr. David Lillis *

Time Allowed: 120 minutes

Instructions for Candidates

Answer any 2 questions. All questions carry equal marks.

BJUT Student ID: _____ **UCD Student ID: _____**

I have read and clearly understand the Examination Rules of both Beijing University of Technology and University College Dublin. I am aware of the Punishment for Violating the Rules of Beijing University of Technology and/or University College Dublin. I hereby promise to abide by the relevant rules and regulations by not giving or receiving any help during the exam. If caught violating the rules, I accept the punishment thereof.

Honesty Pledge：_____ (Signature)

Instructions for Invigilators

No special instructions.

Question 1:

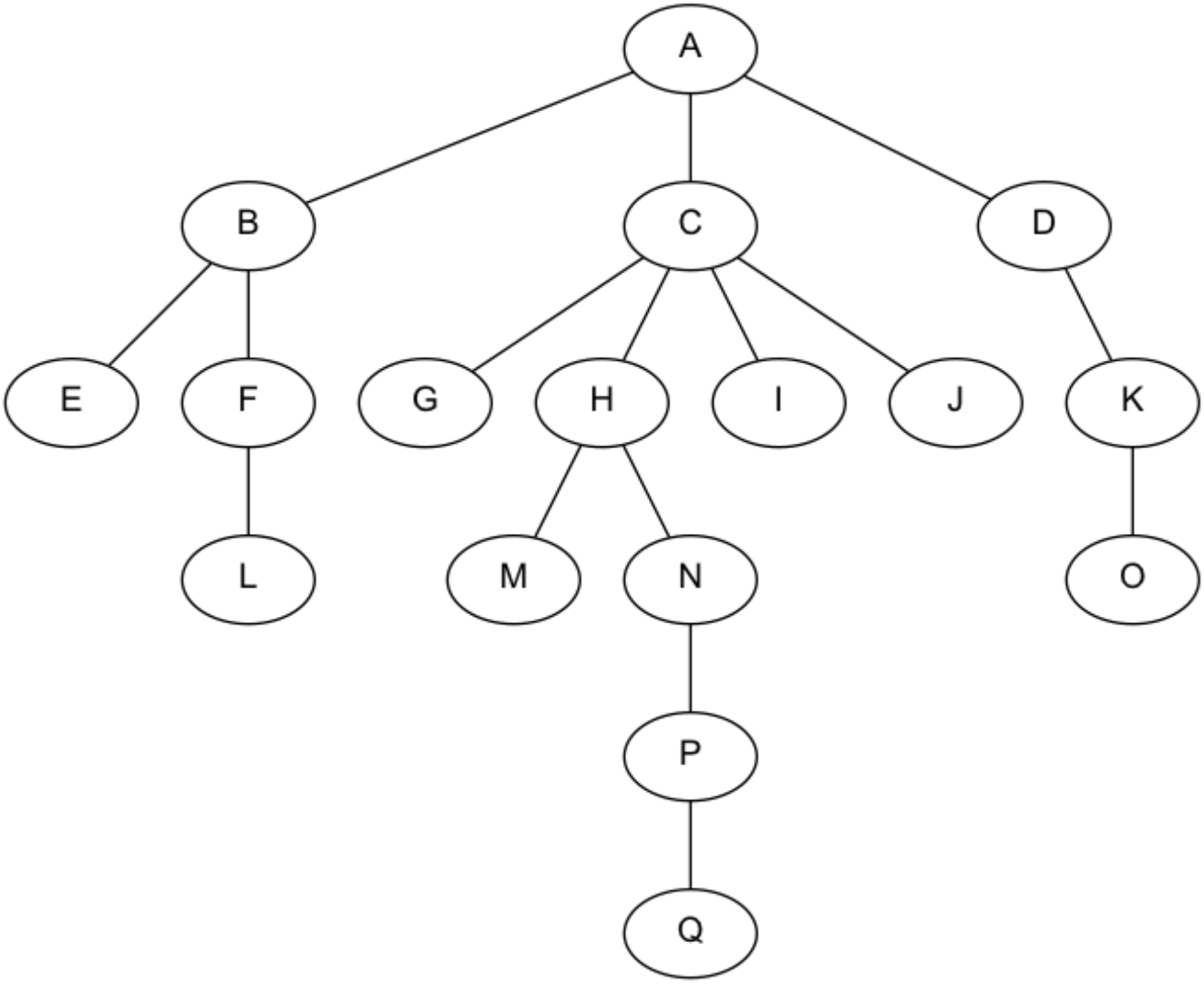

Figure 1

(a) Study the tree in Figure 1 and answer the questions that follow.

(i) List the siblings of F.

(ii) What is the depth of P?

(iii) How many external nodes are in this tree?

(iv) List the ancestors of A.

(v) What is the degree of node H?

(vi) What is the height of the tree?

(vii) List the descendants of B.

(viii) Is (Q,P) an edge? Explain your answer.

(ix) Is (H,N,P,Q) a path? Explain your answer.

(x) List the nodes that are in the subtree that is rooted at H.

[10 marks]

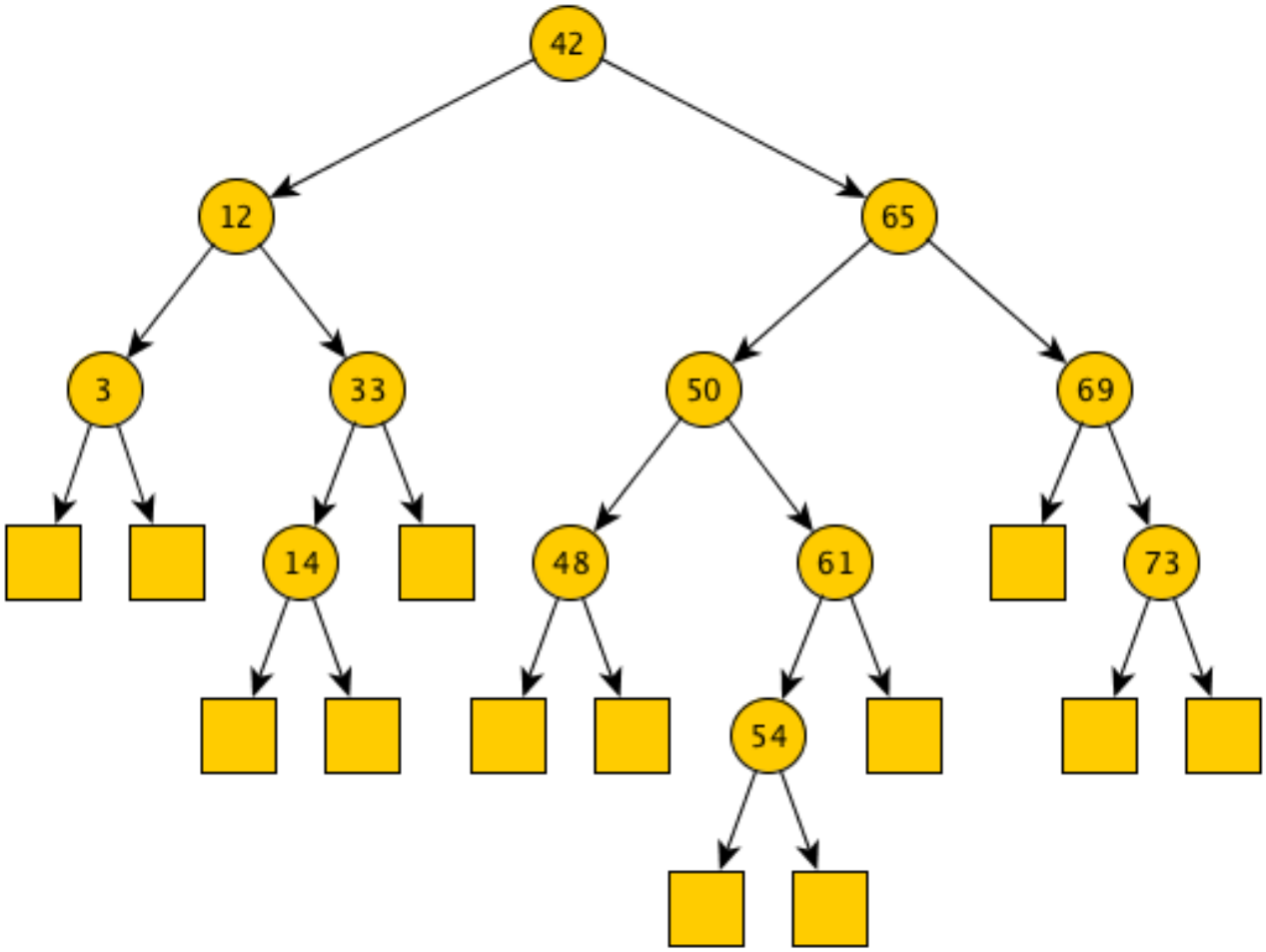

Figure 2

(b) Assume that the tree in Figure 2 is an **AVL Tree**. Draw the state of the tree after performing the following operations. In your answer, you should show the tree's state after each step.

(i) Insert 75

(ii) Remove 3

(iii) Insert 10

(iv) Insert 80

(iv) Remove 50

[12 marks]

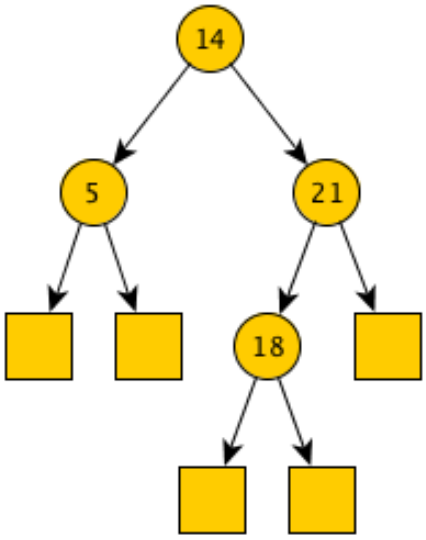

Figure 3

(c) Assume that the tree in Figure 3 is a **Splay Tree.** Draw the state of the tree after performing the following operations. In your answer, you should show the tree's state after each step.

(i) Find 18

(ii) Insert 24

(iii) Insert 16

(iv) Remove 16

[10 marks]

(d) With regard to analysing the performance of a **Binary Search Tree**, discuss each of the following aspects:

(i) Space requirements.

(ii) Time complexity of the insert() and remove() functions.

(iii) The effect that a balanced tree has on the time complexity of the insert() and remove() functions.

[8 marks]

(e) Answer the following questions relating to the traversal of binary trees.

(i) Show the values found during a *postorder* traversal of the tree in Figure 2.

(ii) Show the values found during an *inorder* traversal of the tree in Figure 2.

(iii) The following are two traversals of a binary tree. Draw the tree.

Postorder: D, E, F, G, H, I, J

Inorder: H, G, E, D, F, I, J

[10 marks]

[Total 50 marks]

Question 2:

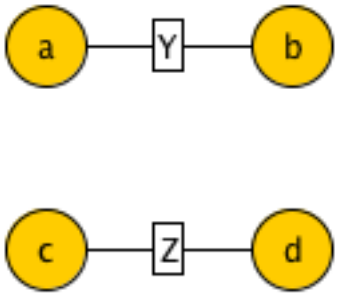

Figure 3

(a) Draw a diagram to show how the graph in Figure 3 can be represented using an *adjacency matrix* structure. In your answer, describe each object type and data structure used to explain its purpose and what data it stores.

[12 marks]

(b) One implementation of a Priority Queue data structure is to use a *Heap*.

(i) Draw the heap that would be obtained after each of the following operations (your answer should show 10 heap diagrams).

insert(20), insert (24), insert(6), insert(14), insert(2),
remove(), remove(), insert(18), remove(), insert(12)

[10 marks]

(ii) A heap is a type of *Complete Binary Tree*. What are the properties of a Complete Binary Tree?

[3 marks]

(iii) Explain in detail how an Array-Based List is a suitable way of implementing a Complete Binary Tree.

[6 marks]

(c) What is *Huffman Encoding* used for? Explain in detail how a *Huffman Tree* is generated. Create a Huffman Tree for the string “assessment is fun” to illustrate your answer.

[10 marks]

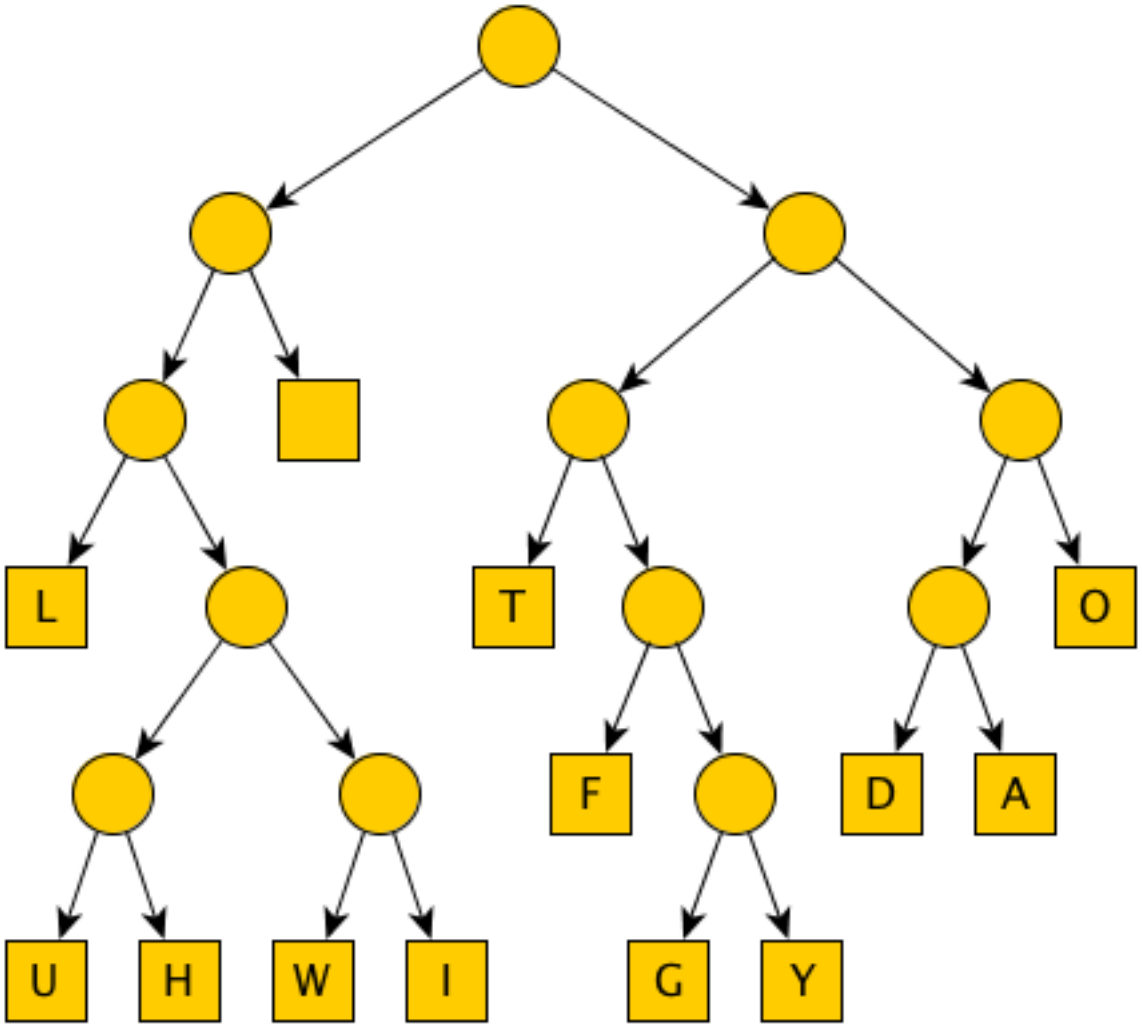

Figure 4

(d) Study the Huffman Tree in Figure 4 and answer the questions that follow. In this tree, the empty square represents a space character.

(i) Using an example from the tree, explain what a *prefix code* is.

[5 marks]

(ii) What text does the following code represent?

101101111111100011100110110111

[4 marks]

[Total 50 marks]

Question 3:

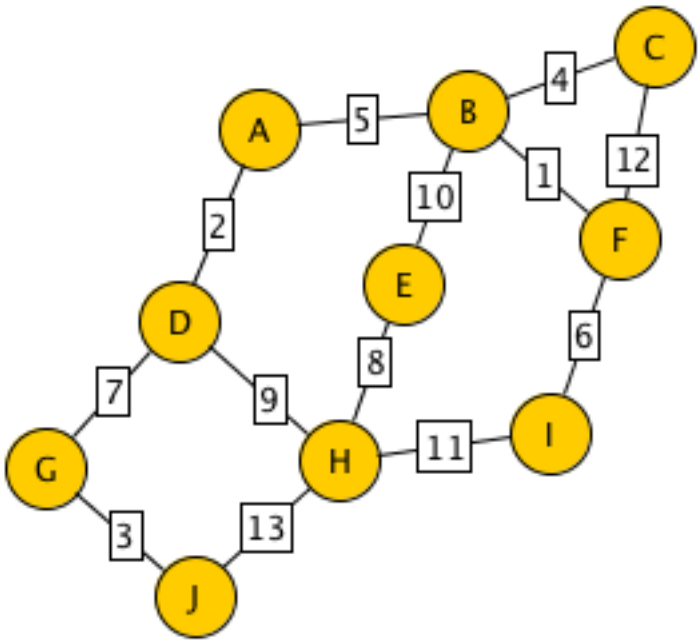

Figure 5

(a) This part relates to *Minimum Spanning Trees*.

(i) What is a Minimum Spanning Tree?

[2 marks]

(ii) Using the graph in Figure 5 as an example, show how Kruskal's algorithm can be used to compute a Minimum Spanning Tree. In your answer, you must explain each step that you take.

[12 marks]

(b) *Dijkstra's Algorithm* computes the distances of all vertices from a given start vertex.

(i) Dijkstra's Algorithm makes some assumptions about graphs it is used with. Name two features a graph may have that would make it unsuitable for using Dijkstra's algorithm.

[4 marks]

(ii) Using the graph in Figure 5 as an example, show how Dijkstra's algorithm can be used to generate a shortest distance tree beginning at vertex "J". In your answer, you must explain each step that you take.

[12 marks]

(c) Explain in detail how a *Depth First Search* graph traversal is done. To illustrate your answer, show how a Depth First Search traversal would work for the graph in Figure 5.

[12 marks]

(d) When implementing a graph, an *adjacency list* structure is an extension of the *edge list* structure.

(i) What are the differences between an adjacency list structure and an edge list structure?

[2 marks]

(ii) What effect do these differences have on the running time complexity of the incidentEdges(v), areAdjacent(v,w) and removeEdge(e) functions? Explain your answer.

[6 marks]

[Total 50 marks]