Duration: **50 minutes**
Aids Allowed: **NONE**

Student Number: |_|_|_|_|_|_|_|_|_|

Last (Family) Name: ______________________________

First (Given) Name(s): ______________________________

---

*Do **not** turn this page until you have received the signal to start.*
(In the meantime, please fill out the identification section above,
and read the instructions below *carefully*.)

This term test consists of 2 questions on 6 pages (including this one). *When you receive the signal to start, please make sure that your copy of the test is complete.*

Answer each question directly on the test paper, in the space provided, if you need more space for one of your solutions, use the reverse side of the final page and *indicate clearly the part of your work that should be marked.*

In your answers, you may use without proof any fact covered in lecture, tutorial, or on assignments. You must justify all other facts required for your solution.

Marking Guide

1: ______/13

2: ______/12

TOTAL: ______/25

Good Luck!

Question 1. [13 MARKS]

Short answer questions. You do not need to justify your answers unless indicated.

Part (a) [1 MARK]

The height a 2-3-4 tree depends on the number of keys in each node. Let h be the height of a 2-3-4 tree with n keys, then $low \leq h \leq high$. What are low and $high$?

Part (b) [1 MARK]

Is there a **best-case** scenario where `INSERT(key k)` for a 2-3-4 tree with n keys has complexity less than the worst-case complexity? In either case, what is this complexity?

Part (c) [1 MARK]

What is the **worst-case** time complexity of `DECREASE-PRIORITY(key k, priority p)` for a min heap of size n?

Part (d) [1 MARK]

Is there a **best-case** scenario where `INSERT(key k)` for a max heap of size n has complexity less than the worst-case? In either case, what is this complexity?

Part (e) [3 MARKS]

The *distance* between any two vertices in a graph is the length of the shortest path between them. The *diameter* of a graph is the maximum distance between two vertices in the graph. Give an algorithm to find the diameter of a graph. State the complexity of your algorithm.

Part (f) [3 MARKS]

Given a weighted graph with n vertices and m edges where all the edge weights are 10, give an efficient algorithm to find the minimum spanning tree. You may refer to any algorithm discussed in class. What is the complexity of your algorithm?

Part (g) [3 MARKS]

Does the order of operations matter when doing a sequence of deletions on a 2-3-4 tree? Justify your answer. (HINT: try deleting keys from a 2-3-4 tree containing 6 keys.)

Question 2. [12 MARKS]

Recall the ADT PLOT from assignment 1. An object of the ADT is a set of data points. Each point P is a co-ordinate (P_x, P_y) in the plane. The operations of the ADT are:

`DELETE`(S, P): Deletes the point P from the set S. If P was not in S then this operation does nothing.

`ADD`(S, P): Adds the point P to the set S.

`MIN_VALUE`(S, x_1, x_2): Returns the minimum value P_y such that $x_1 \leq P_x \leq x_2$ and P is a point in S. If there is no point P such that $x_1 \leq P_x \leq x_2$ then return 0.

In the following questions, you will be given a variation of the PLOT ADT and asked to give an implementation. You will be graded on the space and time complexity as well as the simplicity of your data structure. You may alter data structures from assignments or examples from class or come up with an entirely new one.

Part (a) [6 MARKS]

In the assignment, we assumed that the x co-ordinate for each point is unique (*i.e.,* no two points have the same x value). Now consider when **multiple points** with the same x co-ordinate or y co-ordinate are allowed. Describe a data structure to implement this ADT with the operation `MIN_VALUE`(S, x_1, x_2) replaced by a new operation, `MAX_VALUE`(S, x) described below.

`MAX_VALUE`(S, x): Returns the point P such that $P_x = x$ and P_y is maximum.

Make sure you carefully describe each *operation* and its *complexity* (`ADD`(S, P), `DELETE`(S, P), and `MAX_VALUE`(S, x)). You do not need to give the details of methods discussed in class, however you should state their complexity.

Part (b) [6 MARKS]

Suppose we now *add* the additional operation `GET_MAX_MODE(`S`)` to the ADT of part (a) where `GET_MAX_MODE(`S`)` is defined as follows:

`GET_MAX_MODE(`S`)`: Return the y value such that the number of points in S with $P_y = y$ is maximum.

Describe a data structure to implement this ADT. Make sure you explain how the other operations (`ADD(`S, P`)`, `DELETE(`S, P`)`, `MAX_VALUE(`S, x`)`) are affected. You may alter your data structure from (a) or give an entirely new one.

. . . this page is for extra space.