COMP 271: Mid-term Exam
2002 Fall Semester
2 November 2002
2:00 pm – 4:00 pm

You have 2 hours to solve 5 problems for a total of 100 marks. Answer all problems in the space provided. You may also use the back of the test papers. Good luck!

Student name: ____________________ **Student ID:** ____________________

Problem	Your mark	Maximum
1		10
2		20
3		25
4		20
5		25

Your total marks: ____________

Problem 1. (10 points)

Consider the following recurrence relation:

$$\begin{aligned} T(1) &= 1 \\ T(n) &= 4T(n/2) + n, \qquad \text{for } n > 1, \end{aligned}$$

where n is a power of 2. Solve this recurrence using any method you like. Show all your calculations.

Problem 2. (20 points)

Let $T(n, i)$ denote the average number of comparisons of array elements done by the *Randomized-Select* algorithm for determining the ith smallest of n elements.

(a) (10 points) Write the recurrence relations for $T(n, 1)$, $T(n, 2)$, and $T(n, 3)$.

(b) (10 points) Give the exact values of $T(1, 1)$, $T(2, 2)$, and $T(3, 3)$. Show your calculations.

Problem 3. (25 points)

A *bipartite* graph is an undirected graph $G = (V, E)$ whose vertices can be partitioned into two subsets such that there is no edge between any two vertices in the same subset. (In other words, G is bipartite if and only if there exist two sets V_1 and V_2 such that $V_1 \cup V_2 = V$, $V_1 \cap V_2 = \emptyset$, and all the edges in E connect some vertex in V_1 with some vertex in V_2.)

(a) (3 points) Give a bipartite graph consisting of 5 vertices and 4 edges.

(b) (6 points) Prove that a bipartite graph has no cycle of odd length.

(c) (10 points) Design an algorithm based on breadth-first search (BFS) to determine if an undirected graph G is bipartite.

Note: It is enough to describe the main ideas of your algorithm at a high level in 6-8 lines.

(d) (6 points) Fully justify the correctness of your algorithm.

Problem 4. (20 points)

Let $G = (V, E)$ be a connected undirected graph with weights on the edges. Assume that all the edge weights are distinct. Let $u \in V$ be any vertex and let $e = (u, v)$ be the minimum weight edge among the edges incident on u. Prove that e must be included in any minimum spanning tree of G.

Note: You have to prove this from first principles, i.e., you are not allowed to use the MST Lemma or assume the correctness of Kuskal's or Prim's algorithm.

Problem 5. (25 points)

In the *max-min* problem, you are required to find *both* the largest and smallest number in an array of n values.

(a) (10 points) Show that for $n = 8$, the max-min problem can be solved using at most 10 comparisons.

Note: The naive approach of scanning the array twice, first to find the largest number and then to find the smallest number takes 14 comparisons. For this problem, you need to do something cleverer based on divide-and-conquer.

(b) (15 points) Assume that n is a power of 2. Design a divide-and-conquer algorithm for the max-min problem that uses at most $3n/2 - 2$ comparisons. Also, show that the number of comparisons used by your algorithm is at most $3n/2 - 2$.