Duration: **80 minutes**
Aids Allowed: **NONE** (in particular, no calculator)

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

Last (Family) Name(s): ____________________

First (Given) Name(s): ____________________

---

*Do **not** turn this page until you have received the signal to start.*
In the meantime, please read the instructions below *carefully*.

This term test consists of 3 questions on 9 pages (including this one), printed on both sides of the paper. *When you receive the signal to start, please make sure that your copy of the test is complete, fill in the identification section above, write your student number where indicated at the bottom of every odd-numbered page (except page 1), and write your name on the back of the last page.*

Answer each question directly on the test paper, in the space provided.

In your answers, you may use without proof any result or theorem covered in lectures, tutorials, homework, tests, or the textbook, as long as you give a clear statement of the result(s)/theorem(s) you are using. You must justify all other facts required for your solutions.

Write up your solutions carefully! In particular, use notation and terminology correctly and explain what you are trying to do.

If you are unable to answer a question (or part), you will get 20% of the marks for that question (or part) if you write "I don't know" and nothing else — you will get 10% of the mark if your answer is completely blank. You will *not* get these marks if your answer contains contradictory statements (such as "I don't know" followed or preceded by parts of a solution that have not been crossed off).

Marking Guide

1: ______/12

2: ______/12

3: ______/12

Bonus Marks: ______/ 4

TOTAL: ______/36

Good Luck!

Question 1. Equitable Spanning Trees [12 MARKS]

You are given an undirected graph $G = (V, E)$ with a non-negative weight assigned to each edge $w : E \to \mathbb{R}^+$. In class we saw how to compute the minimum spanning tree of G, that is a spanning tree T of G such that the total weight of the edges of T, $\sum_{e \in E(T)} w(e)$, is minimum. For this question we are interested in a different kind of spanning tree. For any spanning tree T of G we define its "inequity" be the difference between the highest weight edge of T and the lowest weight edge of T. That is,

$$\mathit{inequity}(T) = \left(\max_{e \in E(T)} w(e)\right) - \left(\min_{e \in E(T)} w(e)\right).$$

Devise an algorithm that given a graph G finds a spanning tree of G of minimum inequity. Breifly justify that your algorithm is correct and analyze its runtime. In order to get the full marks your algorithm should run in time at most $O(|E|^2 \log |E|)$.

Example: $G = (V, E)$; $V = \{v_1, v_2, v_3, v_4\}$; $E = \{e_1 = (v_1, v_2), e_2 = (v_2, v_3), e_3 = (v_3, v_4), e_4 = (v_1, v_3)\}$; $w(e_1) = 1$, $w(e_2) = 2$, $w(e_3) = 5$, $w(e_4) = 3$. The graph can be seen in Figure 1.

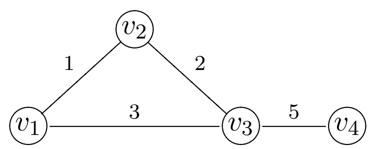

Figure 1: An undirected weighted graph G (Question 1.)

The answer is the tree T with vertices V and edges e_2, e_3, e_4. The inequity of this tree is 3.

[There's more space on the next page...]

Question 1. (CONTINUED)

[Use this page for the rest of your solution.]

Question 2. Fires on the Track [12 MARKS]

You are the head of the fire department responsible for the formula 1 car racing event. The race track is one huge circle of circumference c miles. The northern most point of the track is the starting point of the race and is called point 0 while the position of the other points are defined in terms of how many miles from the starting point they are if one starts from the starting point and drives on the track clockwise.

Unfortunately, there was an accident there and some intervals of the track are on fire. To be precise there are n closed intervals $[a_i, (a_i + l_i) \mod c]$ of the road that are now on fire; in other words the ith interval starts at point a_i and is of lenght l_i. See the example below. You can assume that $l_i < c$.

You have an airtanker (an aircraft that dumps water on fires) that you can use to put out the fires. In each flight the airtanker can fly above any point of the track and start dumping water while continuing to fly over the road in the clockwise direction. Because the capacity of the tanker and its flying speed are limited and the track is a circle if the airtanker starts dumping water at mile x of the road it runs out of water at mile $(x + t) \mod c$. In other words in each flight the airtanker will dump water over a half open interval of the road of the form $[x, (x + t) \mod c)$. Any fire at any point of this interval is put out.

You want to put out all the fires with the *minimum number of flights*. Devise an algorithm that takes c, n, t, the starting points of the fire intervals $a_1, \ldots, a_n$ and their lengths $l_1, \ldots, l_n$ as input and finds the minimum number of flights it takes to put out all the fires. You can assume that the fire intervals $[a_i, b_i]$ do not intersect and that $0 \le a_i < c$ for all i. Briefly justify that your algorithm is correct and analyze its runtime. In order to get the full marks your algorithm should have runtime at most $O(n^2)$.

Example: If $c = 20$, $n = 4$, $t = 4$, $a_1 = 2$, $a_2 = 10$, $a_3 = 12$, $a_4 = 19$, $l_1 = 1$, $l_2 = 1$, $l_3 = 1$ and $l_4 = 2$. This input is drawn in Figure 2.

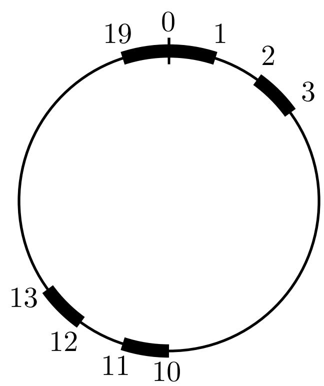

Figure 2: An example of fires on the track. The track is drawn with a line, the fires with very thick lines. (Question 2.)

The optimal number of flights is 3. One optimal solution that achieves this number of flights is to dump water on the intervals $[18.5, 2.5)$, $[2.5, 6.5)$ and $[10, 14)$.

[There's more space on the next page...]

Question 2. (CONTINUED)

[Use this page for the rest of your solution.]

Question 3. Numbers on a tree [12 MARKS]

You have a rooted tree T with n vertices with a number a_i written next to the vertex i. You want to find numbers $b_1, \ldots, b_n$ and write b_i in place of a_i so that they satisfy the following property. The number written next to vertex i should be at least as big as the sum of the numbers written next to all the children of the vertex i. In other words for all i,

$$b_i \geq \sum_{j:\text{vertex } j \text{ is a child of vertex } i} b_j.$$

All the numbers $a_1, \ldots, a_n$ are between 0 and l and the numbers $b_1, \ldots, b_n$ should also be in the same interval. Define the *deviation* of b_i's as the maximum distance between an a_i and its corresponding b_i, i.e.,

$$\text{deviation} = \max(|a_1 - b_1|, |a_2 - b_2|, \ldots, |a_n - b_n|).$$

Design an algorithm that finds the b_i's with the *minimum deviation* in runtime $O(nl)$. Briefly justify that your algorithm is optimal and runs in the correct runtime. To make the problem easier you can assume that if vertex j is a child of vertex i then $j > i$ and vertex 1 is the root of the tree. Your algorithm is given n, l, $a_1, \ldots, a_n$ and $C_1, C_2, \ldots, C_n$ as input, where C_i is the vertices which are the children of vertex i, and has to output the optimal $b_1, \ldots, b_n$.

Example: The tree for $n = 5$, $l = 10$, $a_1 = 1$, $a_2 = 4$, $a_3 = 1$, $a_4 = 3$, $a_5 = 2$ and $C_1 = \{2, 3\}$, $C_2 = \{4, 5\}$, $C_3 = \varnothing$, $C_4 = \varnothing$, $C_5 = \varnothing$ is drawn in Figure 3.

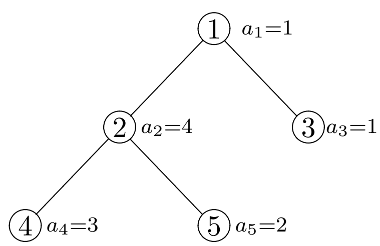

Figure 3: An example for Question 4.

The minimum deviation is 2. An optimal answer is $b_1 = 3$, $b_2 = 3$, $b_3 = 0$, $b_4 = 2$, $b_5 = 0$.

Bonus (4 extra marks): Design an algorithm that runs in time $O(n \log l)$.

[There's more space on the next page...]

Question 3. (CONTINUED)

[Use this page for the rest of your solution.]

Last (Family) Name(s): ______________________________

First (Given) Name(s): ______________________________