

**Due:** Tuesday June 5th at 6:00PM in class, or Thursday June 7th at 6:00PM electronically

**Worth:** 4%.

For each question you have to clearly write your algorithm in English and prove that it finds the optimal answer. In any of your answers you can use any algorithm we discussed in class without proving it solves the problem we discussed in class optimally. If we discussed the runtime of the algorithm you can also use that without reproving it. The same goes for any Lemma, Theorem or Fact we discussed in class.

### Question #1: High Speed Trains (2pt)

---

You are given an undirected graph  $G = (V, E)$  with a non-negative number assigned to each edge  $t : E \rightarrow \mathbb{R}^+$ . The nodes of the graph are the cities in Canada and each edge is a route along which a high speed train track is planned to be constructed. The number  $t((u, v))$  is how many years from now the (two-directional) direct high speed route between cities  $u$  and  $v$  is going to be finished.

Devise the fastest algorithm that given  $G$  and  $t$  computes the earliest time, in number of years from now, that one can travel between all cities in Canada on high speed trains.

Hint: Minimum Spanning Trees might be useful.

**Example:**  $G = (V, E); V = \{v_1, v_2, v_3, v_4\}; E = \{e_1 = (v_1, v_2), e_2 = (v_2, v_3), e_3 = (v_1, v_4), e_4 = (v_1, v_3)\}; t(e_1) = 2, t(e_2) = 2, t(e_3) = 1, t(e_4) = 3.$

The answer is 2 because two years from now  $e_1, e_2$  and  $e_3$  are constructed and one can travel between any two cities using them and before that point it is impossible to travel between  $v_1$  and  $v_2$ .

### Question #2: Fires on the Road (2pt)

---

You are the head of the fire department in a sparsely populated area of northern Canada. There is a straight south-north road of length  $l$  miles in the area you are responsible for. Unfortunately, there was an accident there and some intervals of the road are on fire. To be precise there are  $n$  closed intervals  $[a_i, b_i]$  of the road that are now on fire. 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 road and start dumping water while continuing to fly north over the road. Because the capacity of the tanker and its flying speed are limited if the airtanker starts dumping water at mile  $x$  of the road it runs out of water at mile  $x + t$ . In other words in each flight the airtanker will dump water over a half open interval of the road of the form  $[x, x + t)$ . Any fire at any point of this interval is put out.

You are given  $l, n, t$  and the intervals  $[a_1, b_1], [a_2, b_2], \dots, [a_n, b_n]$  and you want to put out all the fires with the *minimum number of flights*. Give the fastest solution you can think of and analyze its run time. You can assume that the intervals  $[a_i, b_i]$  do not intersect and that  $0 \leq a_i \leq b_i \leq l$  for all  $i$ .

**Example:** If  $l = 10, n = 3, t = 3$  and the 3 intervals are  $[1, 5], [6, 6]$  and  $[10, 10]$  the optimal number of flights is 3. One optimal solution that achieves this number of flights is to dump water on the intervals  $[0.5, 3.5), [3.5, 6.5)$  and  $[10, 13)$ .

**Question #3: Making Change (2pt)**

---

In Canada we have coins. You are working as a cashier at a popular grocery store and someone just paid with a hundred-dollar bill.<sup>1</sup> You have to give them back  $x\text{¢}$  worth of change using only Canadian coins.<sup>2</sup> But as any cashier will tell you coins are always in short supply in grocery stores so you want to do that using the *minimum number of coins*. Assume that you have ample supply of all Canadian coins, 1¢, 5¢, 10¢, 25¢, \$1 and \$2.

Given  $x$  come up with the fastest algorithm that computes the minimum number of coins required and analyze its run time.

**Example:** If  $x = 126$  the answer is 3. You can use one \$1, one 25¢ and one 1¢ coin to return the change.

**Question #4: Numbers on a path (3pt)**

---

You are given  $n$  integers  $a_1, \dots, a_n$  all between 0 and  $l$ . Under each integer  $a_i$  you should write an integer  $b_i$  between 0 and  $l$  with the requirement that the  $b_i$ 's form a non-decreasing sequence, i.e.  $b_1 \leq b_2 \leq \dots \leq b_n$ . You define the *deviation* of  $b_i$ 's as the maximum distance between an  $a_i$  and its corresponding  $b_i$ , i.e.,

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

Design an algorithm that finds the  $b_i$ 's with the *minimum deviation* in runtime  $O(n\sqrt[4]{l})$ .

Hint 1: First try to find an algorithm with runtime  $O(nl)$ .

Hint 2: The actual runtime of the optimal algorithm is much less than  $\Theta(n\sqrt[4]{l})$ .

**Example:** If  $n = 2$ ,  $l = 10$ ,  $a_1 = 4$ ,  $a_2 = 2$  then the minimum deviation is 1. An optimal answer is  $b_1 = 3$ ,  $b_2 = 3$ .

---

<sup>1</sup>isn't that annoying?

<sup>2</sup>to get back at them of course.