Homework 4

CS 577
Jason Jiang

February 20, 2019

1. The year is 1922. You are organizing a big dance in your university and you want to invite everyone. Unfortunately, emails and cellphones were not available that time so you have to visit all classes during the lectures and invite the students of each class to the event. The lecture time of each class is a half-closed interval (si; fi] of a start and nish time. To minimize your trips to the university you try to visit it during times where many classes overlap and invite the students of all these classes. For simplicity we assume that the start 1 and nish times of the lectures are non-negative rational numbers. One possible schedule of lectures could be

$$(1, 3.1]; (2.2, 4]; (1.7, 5]; (4, 5]$$

In this case the smallest number of visits to the university is 2, for example one visit at 2:5 and one at 4:5. Visiting at 4 does not cover the (4, 5] class, however does cover the (2.2, 4] class.

a)You decide that you should pick your first visit to be a time that overlaps with the maximum number of lectures, that is a number t that is contained in the maximum number of intervals (si; fi]. Then you remove these intervals and pick your second visit with the same rule. You continue until no more lectures are uncovered. Construct a counter-example where this greedy strategy fails to compute an optimal solution.

b)Design a greedy algorithm that computes the smallest number of visits to the university that cover all lectures and runs in time O(n log n).

- In order to find a counter-example, we have to find a scenairo that picking a time with not the maximum number of overlaps that will eventually lead to a better result. Say, here is an example:

$$(1, 3], (2, 5], (2, 5], (4, 7], (4, 7], (6, 8]$$

Given those 6 time intervals, by the naive greedy alogrithm, we should pick (4,7], (4,7], (2,5], (2,5] at the first time, say we pick 4 lectures at the same time. Then we pick (1,3] for the second time, and finally, (6,8] for the last time. Total 3 visits.

However, if we go (1,3], (2,5] and (2,5] at the first time, with 3 lectures at the same time, and the next one we just go (4,7], (4,7] and (6,8] for the second time then we are done. Total is just 2 visits.

- **Input:**
  Output:
  procedure COUNT_MIN_VISIT()