CS 3000: Algorithms & Data — Summer 1 '20 — Tim LaRock

Extra Credit Assignment 2
Due Thursday June 18th 5PM via Gradescope

Name:
Collaborators:

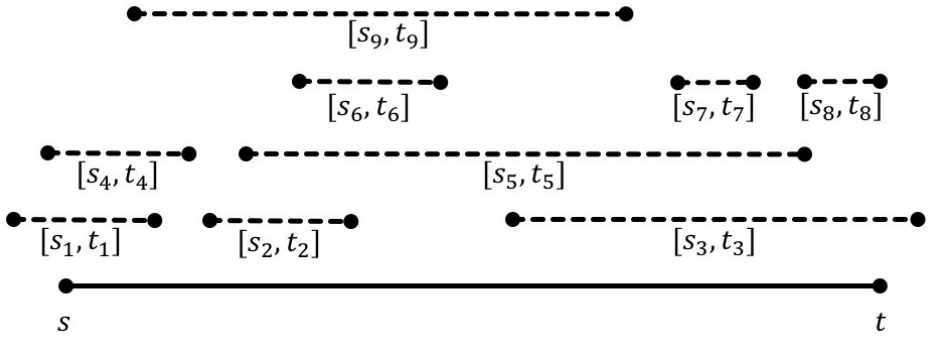
- This is an extra credit assignment. Up to 10 points earned on this assignment will be added to your lowest homework grade (after dropping your lowest-lowest grade).
- Make sure to put your name on the first page. If you are using the LaTeX template we provided, then you can make sure it appears by filling in the `yourname` command.
- This assignment is due Thursday June 18th 5PM via Gradescope. Make sure to submit something before the deadline.
- Solutions must be typeset in LaTeX. If you need to draw any diagrams, you may draw them by hand as long as they are embedded in the PDF. I recommend using the source file for this assignment to get started.
- I encourage you to work with your classmates on the homework problems. *If you do collaborate, you must write all solutions by yourself, in your own words.* Do not submit anything you cannot explain. Please list all your collaborators in your solution for each problem by filling in the `yourcollaborators` command.
- Finding solutions to homework problems on the web, or by asking students not enrolled in the class, is strictly forbidden.

Problem 1. *Greedy Algorithms*

You are running a convention on a shoestring budget, and need to staff the registration desk. The convention runs for the interval $[s, t]$. There are n potential staff, each of which is able to cover an interval $[s_i, t_i]$. You need to select a set of volunteers $S \subseteq \{1, \ldots, n\}$ to *cover* the entire convention, meaning that the union of all of their intervals covers the entire interval, i.e. $\bigcup_{i \in S} [s_i, t_i] \supseteq [s, t]$. Equivalently, for every time $z \in [s, t]$, there is some volunteer $i \in S$ such that $z \in [s_i, t_i]$. However, each staffer will be paid for their services out of very limited funds, so you need to ensure $|S|$ is as small as possible.

In this problem you will design an efficient greedy algorithm that takes as input the numbers $s, t, s_1, t_1, \ldots, s_n, t_n$ and outputs a set S that covers the convention and uses the minimum number of staffers. The running time of your algorithm should be at most $O(n^2)$, but a solution exists that runs in $O(n \log n)$ time.

The following is an example input with 9 volunteers. One optimal solution is $S = \{1, 3, 9\}$.



(a) *(1 point)* In a few sentences, explain how your algorithm will work.

Solution:

(b) *(1 point)* Describe your algorithm in pseudocode.

Solution:

(c) *(1 point)* Analyze the running time of your algorithm.

Solution:

(d) *(2 points)* Prove that your algorithm finds a set of staffers S of minimum size to cover the convention. Note: A greedy-stays-ahead proof will probably be easier.

Solution:

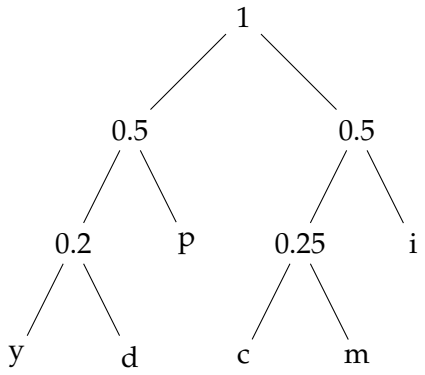
Problem 2. *Huffman Codes*

You are given the symbol alphabet $\Sigma = \{a, b, c, d, e, f, g\}$ and the following frequencies f_i for each symbol $i \in \Sigma$.

Symbol	a	b	c	d	e	f	g
Frequency	0.25	0.22	0.16	0.14	0.13	0.07	0.03

(a) *(2 points)* Use Huffman's Algorithm to create a prefix-free binary code from the given alphabet and frequencies. You do not need to show your work, but you should format your binary tree in LaTeX starting from the template below. Note that the data in the template are only for illustrative purposes and have nothing to do with the actual solution.

Solution:



(b) *(1 point)* Encode the string *aced*

Solution:

(c) *(1 point)* Decode the following encoded string: 0100100010001001100

Solution:

(d) *(1 point)* Compute the *entropy* $H = \sum_{i \in \Sigma} f_i \cdot \log_2(f_i)$ of the code. Note that entropy is a negative quantity.

Solution: