CS 3000: Algorithms & Data — Spring '20 — Jonathan Ullman

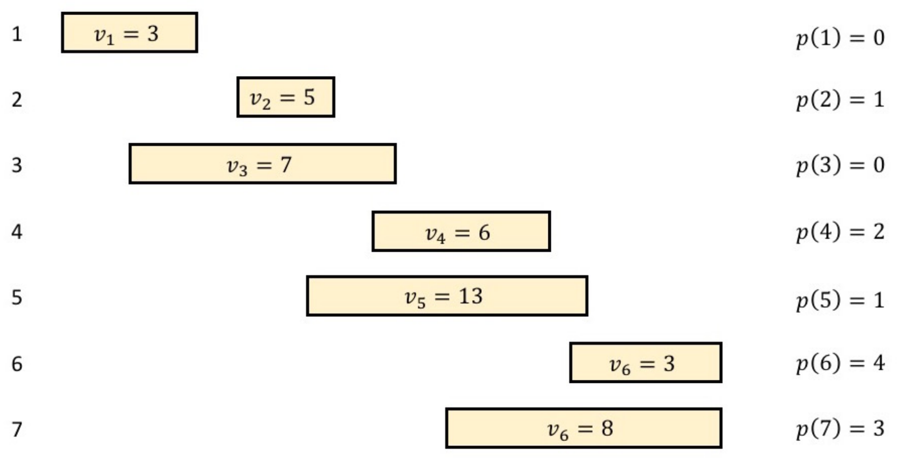
Homework 3
Due Friday January 31 at 11:59pm via Gradescope

Name:
Collaborators:

- 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 Friday January 31 at 11:59pm via Gradescope. No late assignments will be accepted. 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. *Interval Scheduling Recap*

This problem will test your understanding of dynamic programming by having you run through the algorithm for interval scheduling that we saw in class. Consider the following input for the interval scheduling problem:



Fill out the dynamic programming table with the values of $\textsc{opt}(i)$ for $i = 0, 1, \ldots, 7$. Label each value of $\textsc{opt}(i)$ with whether or not the optimal schedule of length i contains interval i or not. Write the optimal schedule and its value.

Solution:

Problem 2. *Concert Calendar*

You are a lover of live music, and each weekend you have to choose a concert to go to. Every weekend you have the choice of going to the Paradise Rock Club to see Scandinavia's finest epic metal or to swing by the Scullers for some low key jazz. You generally prefer epic metal, but those concerts are loud, and if you go to one, you will have to take the next weekend off to recover. The goal is to take the concert calendar, and decide each week whether to see an epic metal show or an adult contemporary show.

On weekend i, if you go to a jazz concert, you get value $j_i > 0$, and if you go to a metal concert, you get value $m_i > 0$. However, if you go to a metal show, then you will not be able to go to any concert on weekend $i+1$. Given a set of values $j_1, m_1, \dots, j_n, m_n$ for n weekends, a *schedule* is a set of choices for each of the n weekends, where each weekend you can either choose "metal," "jazz," or "none," and if you choose "metal" in weekend i then you must choose "none" in weekend $i+1$. The *value* of a schedule is the sum of the values of the concert you choose on each weekend.

Your goal is to design a dynamic programming algorithm that takes as input the n values $j_1, m_1, \dots, j_n, m_n$ and computes an *optimal* schedule (i.e. a schedule with maximum value).

For example, if given the input:

	Weekend 1	Weekend 2	Weekend 3	Weekend 4
j	3	6	5	4
m	7	12	8	6

the optimal schedule would be to go to the jazz concerts on weekend 1, the metal concerts on weekends 2 and 4, and none on weekend 3, which has a value of $3 + 12 + 6 = 21$.

(a) Let $\text{OPT}(i)$ be the maximum value you can get for going to concerts on weekends $i, i+1, \dots, n$. Give a recurrence to compute $\text{OPT}(i)$ from the values $\text{OPT}(i+1), \dots, \text{OPT}(n)$, and write the base case(s) for this recurrence. Write a few sentences explaining why your recurrence is correct.

Solution:

(b) Using your recurrence, design a dynamic programming algorithm to output the optimal schedule of concerts to go to. You may use either a top-down or bottom-up approach. Remember that your algorithm needs to output the optimal schedule, not only its value.

Solution:

(c) Analyze the running time and space usage of your algorithm.

Solution: