University of Iowa | Homework 9 | CS 3330
Department of CS | Fall 2024

Assigned: Friday 11:59 PM, November 8, 2024
Due: Friday 11:59 PM, November 15, 2024

1. **[A Human Compiler]**: A computer scientist must understand how algorithms work. Your job here is to execute variants of the weighted interval scheduling problem and produce results. We are given the following list of intervals for the weighted interval scheduling problem.

$$[(5,9,3),(3,5,1),(1,4,5),(0,6,8),(4,7,4),(6,10,2),(8,11,4),(3,8,6)]$$

The elements in the list are in the form (s_i, f_i, w_i), where s_i, f_i and w_i are the start time, finish time and weight of job i.

(a) **[5 points]** In the following table, write down the jobs in an increasing order of f_i and compute $p(i)$ for each job i.

Job number	s_i	f_i	w_i	$p(i)$
1				
2				
3				
⋮	⋮	⋮	⋮	⋮
8				

(b) **[5 points]** Draw the recursion tree for this input when you run the brute force (non memoized) version of the weighted interval scheduling algorithm (See slide 11 in the slide 06DynamicProgrammin1.pdf).

(c) **[5 points]** Run the Botom up (memoization) version of the weighted interval scheduling algorithm (See slide 18 in the slide 06DynamicProgrammin1.pdf). Fill in the following table at the end of each iteration.

Iteration	The values in M
1	
2	
3	
⋮	⋮
8	

(d) **[5 points]** Now execute the Find-Solution algorithm (See slide 17 in the slide 06DynamicProgrammin1.pdf) based on your table in 1 (c). Generate the resulting recursion tree. Note that your recursion tree will be a 'chain' graph. Assume you call FIND-SOLUTION(8).

2. **[Not Too Many Items]:** Consider the following variant of the knapsack problem. We are given n items. Each item i has a value v_i and weight w_i. We are also given a positive integer weight limit W for the knapsack and an item budget $K \leq n$. The goal here is to pack the knapsack such that we

select the items which maximize total value, while ensuring that the total weight of the items is less than W and at most K items are selected. Give a dynamic programming algorithm for this problem.

[Hint: Memoization table needs to be three dimensional]

(a) **[10 points]** Write the bellman equation for this problem to demonstrate how this problem could be solved given solutions to smaller instances of the same problem. Include the base cases.

(b) **[10 points]** Design a Top-Down algorithm using memoization to solve the problem above based on your answer to 2(a). Provide a pseudocode.

3. **[Candy Sharing]**: Alice and Bob gathered many chocolate bars during Halloween and want to divide them between themselves. They need our help to create two subsets, one for Alice and one for Bob, in such a way that the sums of segments for each subset, denoted as S_A for Alice and S_B for Bob, are as close to each other as possible.

   Each chocolate bar is associated with a number in the set $\{1, 2, \ldots, k\}$, representing the number of its segments. The objective is to ensure that the sums S_A and S_B are as close to each other as possible.

   (a) **[5 points]** Formulate the problem as an optimization problem.

   (b) **[5 points]** Instead of calling the sets S_A and S_B, let us refer to them as $S_{\min}$ and $S_{\max}$, where $S_{\min}$ is the set with the smaller number of segments. What is the maximum value that $S_{\min}$ can have?

   (c) **[15 points]** Use the dynamic programming paradigm to find $S_{\min}$.

   (d) **[5 points]** What is the complexity of your algorithm? Justify your answer.

4. **[Stardrop Valley]** is a popular farming game for some reason! In this game, you're in charge of a piece of land where you can build different types of farms. The time it takes to harvest crops depends on the soil conditions in different parts of the land. Imagine there are two kinds of farms: a square carrot farm and a rectangular wheat farm.

   Imagine the land as a grid, similar to a checkerboard, where each square is assigned a number representing its fertility. The fertility of a farm is the sum of the numbers associated with the squares it covers. Our aim is to find the optimal arrangement of farms on the land to maximize crop yield in the least amount of time.

   (a) **[10 points]** Let's begin by choosing a rectangle for the wheat farm. Develop an algorithm utilizing the dynamic programming paradigm to maximize the fertility of the farm.

   (b) **[5 points]** What is the complexity of your algorithm. Justify your answer.

   (c) **[10 points]** Next, let's choose a square for the carrot farm. The carrot farm should not overlap with the wheat farm. Develop an algorithm utilizing the dynamic programming paradigm to maximize the fertility of the farm.

   (d) **[5 points]** What is the complexity of your algorithm. Justify your answer.