PSU: CMPSC465

Assignment 1 - Binary Search

Assignment 2 - Dynamic Programming

Assignment

Assignment 2 - Dynamic Programming

Implement Dynamic Programming Algorithm for Independent Set (IS) on a Line Graph

This problem is exactly the one discussed in lecture on 09/25. You are given a graph on n vertices, where the vertices are connected together in a line. Each vertex has a weight. These weights are specified by v_1,v_2, ..., v_n. An independent set (IS) is a subset of the vertices such that no two vertices in the set have an edge between them. The weight of an IS is the sum of the weights of the vertices in the set. The problem is to find a maximum weight IS in the graph.

Recall from the recitation that the dynamic programming recurrence for the problem is given as follows:
OPT(i) = max(v_i + OPT(i - 2), OPT(i - 1))

In this problem, your goal is to implement a bottom-up dynamic programming algorithm, based on the above recurrence. Starter code is provided.

Input: The input is given in a file called input.txt. It contains a list of white-space delimited integers. This list corresponds to the weights of the problem: v_1, ..., v_n.

Output: Your output must contain three lines. The first line should contain a list of white-space delimited integers. These integers should correspond to the values OPT(1), OPT(2), ..., OPT(n), in that order. The second line should contain a single integer, which is the the maximum weight of an IS. The third line should contain a list of white-space delimited integers, sorted in increasing order. This line corresponds to the indices of a maximum weight IS, using a 0-based offset. In the case there is more than one maximum weight IS, you can output an arbitrary one.

Example :

Input: 8 3 7 10 4
Output:
8 8 15 18 19
19
0 2 4

Explanation: There are 5 vertices, with weights 8, 3, 7, 10, 4. The maximum weight IS is given by choosing element 0 (whose weight is 8), element 2 (whose weight is 7), and element 4 (whose weight is 4). The total weight is 19.

Grading:
There will be no partial credit in this assignment, but you will be able to see whether your submission passes the grader's test as soon as you submit. You will know that your submission passes the test if you see the message "SUCCESS: your submission output is correct." If it does not, you are allowed to fix the problem and resubmit, as many times as you want. There will be no additional tests besides the one that is run when you submit.

Note: If you receive a "Killed" error message, that means your program was taking too long to run and was killed by the server (maybe you had an infinite loop somewhere?)

Submission

Details	
Submission count:	None
Submission deadline:	Oct 09 2015 16:40:59 EDT

My Work (main.php?m=editor&asnid=2995&stepid=2996)

Grading

Grading not started

The assignment will be graded after submission .