

CSE 3318 Lab Assignment 3

Due November 8

Goals:

1. Understanding of dynamic programming.
2. Understanding of subset sums.

Requirements:

1. Design, code, and test a C program that uses *dynamic programming* to determine *two* separate subsequences of the input such that the first subsequence sums to the first target value and the second subsequence sums to the second target value.

The input should be read from standard input (which will be one of 1. keyboard typing, 2. a shell redirect (<) from a file, or 3. cut-and-paste. Do NOT prompt for a file name!). The first line of the input will give n , the length of the sequence, along with the two target values. Each of the remaining input lines will include one sequence value. All values will be positive integers.

Your program should echo the target values and the input sequence. If a problem instance has a solution, each of the two subsequences should be output. Each value is to be preceded by its index in the input. A message should be provided for instances without a solution.

2. Submit your program on Canvas by 5:00 pm on Tuesday, November 8. Comments at the beginning of the source file should include: your name, your ID number, and the command used to compile your code on Omega (5 point penalty for non-compliance).

Getting Started:

1. Unlike the *one-dimensional* situation for ordinary subset sums (Notes 07.F), this problem is *two-dimensional*. The same concept of the cost function value being an index to the input S values is to be used. Similarly, as the backtrace moves through the cost matrix, the indices to S will be in strictly descending order.
2. Dynamic programming is the only acceptable method for doing this lab. Do not use a greedy approach! Finding a DP solution for the first target value and then finding a DP solution for the second target value (using leftover inputs) IS WRONG and will receive no points.