## CSE 2320 Lab Assignment 2

Due March 24, 2016

## Goal:

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

## **Requirements:**

1. a. Design, code, and test a C program that uses *dynamic programming* to solve the subset sum problem *with multiplicities*. This extends the subset sum problem in Notes 07 by allowing an S<sub>j</sub>

value to be used up to a specified number of times (the multiplicity  $M_i$  of  $S_i$ ).

b. 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 first have n, the length of the sequences for S and M, followed by the target value m. Each of the remaining lines will include a positive  $S_i$  value and its multiplicity  $M_i$ . These pairs will be

in *strictly ascending* order by  $S_{i}$ .

c. The dynamic programming cost table should be printed only in those cases where m does not exceed 50. This may be done like the provided

http://ranger.uta.edu/~weems/NOTES2320/subsetSum.c, but must also include the number of occurences of the  $S_i$  that is used to reach a value.

d. If the target value *m* cannot be reached, then print a simple message. If a solution exists, a backtrace of the needed  $S_i$  values (and the number of occurrences for each) should be printed.

2. Submit your C program on Blackboard by 3:15 p.m. on March 24. Comment lines should include 1) the compilation command used on OMEGA and 2) the asymptotic worst-case time for the DP computation.

## **Getting Started:**

- 1. Dynamic programming is the only acceptable method for doing this lab. You will need to be careful to assure that your approach is guaranteed to find a solution if one exists.
- 2. Arrays are to be dynamically allocated.
- 3. You may modify existing code (e.g. subsetSum.c).