CSL 356: Analysis and Design of Algorithms
Instructor: Ragesh Jaiswal

1. There is an $n \times n$ grid of one-way street network. At any intersection, you may either travel from east to west or north to south. In how many different ways can you travel from the north-west corner to the south-east corner. Write a program to determine the number of different ways.

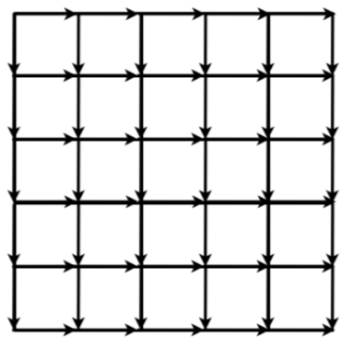

Figure 7.0.1: Example $n = 6$. In how many ways can you go from the top-left corner to the bottom-right corner?

2. You are given n items and a sack that can hold at most W units of weight. The weight of the i^{th} item is denoted by $w(i)$ and the value of this item is denoted by $v(i)$.The items are indivisible. This means that you cannot take a fraction of any item. Design an algorithm that determines the items that should be filled in the sack such that the total value of items in the sack is maximized with the constraint that the combined weight of the items in the sack is at most W. You may assume that all the quantities in this problem are integers.

3. You are given n types of coin denominations of values $v_1 < v_2 < ... < v_n$ (all integers). Assume $v_1 = 1$, so you can always make change for any amount of money C. Give an algorithm that makes change for an amount of money C with as few coins as possible.

4. You have a set of n integers each in the range $0, ..., K$. Partition these integers into two subsets such that you minimize $|S_1 - S_2|$, where S_1 and S_2 denote the sums of the elements in each of the two subsets.

5. Consider a row of n coins of values $v_1, ..., v_n$, where n is even. We play a game against an opponent by alternating turns. In each turn, a player selects either the first or last coin from the row, removes it permanently, and receives the value of the coin. Determine the maximum possible amount of money we can definitely win if we move first.