

**III. Coin Change Problem (Homework 5 Problem 5)**

Suppose that the denominations of the coins in a country are  $d_1 < d_2 < \dots < d_n$  (e.g., 1, 5, 10, 25 for the United States). The problem to consider is:

Given an integer  $M$ , what is the minimum number of coins needed to make  $M$  cents in change? Give a dynamic programming algorithm to solve the problem and analyze its time complexity.

**Solutions.**

1. Learn from rod-cutting

Let's define  $c[j]$  = the minimum of coins needed to make change for  $j$  cents.

After deciding the first coin (similar to left-most cut in rod-cutting), we would make change for the remaining amount in the optimal way by enumerating all the possibilities.

Recurrence:

Python code for demonstration

Example:

$d = [1, 5, 10, 25]$

$M = 30$

## 2. Learn from 0-1 knapsack

Let  $m[i, j]$  = the minimum number of coins needed to make change for  $j$  cents *using any of the coins*  $d_1, d_2, \dots, d_i$ .

- Case 1:  $j < d_i$   
Can we choose coin  $d_i$ ?
- Case 2:  $j \geq d_i$   
Can we choose coin  $d_i$ ?

Recurrence:

Python code for demonstration

Example:

$d = [1, 5, 10, 25]$

$M = 30$