Recitation (IV): Dynamic Programming

Ex 1: Counting Number of Domino Tilings.

Input: \( N \)

Output: Number of ways to cover \( N \times 2 \) grid using domino (1x2 rectangle) tiles.

Examp: \( N = 4 \)
DP Alg

- Subproblem?
  - Consider the direction of the 1st domino.

- Let $\text{opt}[i] := \# \text{ of ways to tile } i \times 2 \text{ grid}$

\[ \Rightarrow \text{opt}[i] = \text{opt}[i-1] + \text{opt}[i-2] \]

- $1^{st}$ domino horizontal
- $1^{st}$ domino vertical
Ex 2: Max-weight Indep Set on Trees

Input: a tree of n nodes, each node i has weight \( w_i \).

Output: An indep set of the tree with the largest weight.

Example:

```
      15
     /   \
   8     16
  /   \   /  \
3     5   7   9
 / \   / \ / \   \
4   5   2  
```

Total weight = 47
DP Alg

- Sub problem?

\[ \text{opt}[U, 1] := \max \text{ wt of indep set rooted at vertex } U, \text{ but does not include } U. \]

\[ \text{opt}[U, 2] := \max \text{ wt of indep set rooted at vertex } U, \text{ but does include } U. \]

\[ \Rightarrow \text{ opt}[U, 1] = \sum_{U: \text{child of } V} \max \{ \text{opt}[U, 1], \text{opt}[U, 2] \} \]

\[ \text{opt}[U, 2] = W_v + \sum_{U: \text{child of } V} \text{opt}[U, 1] \]
Ex 3: Coin Change

Input: Seq of int $C_1 < C_2 < C_3 < \cdots < C_n$ as coin types.

Integer $M$

Output: # of distinct ways to make a $M$-cents change, assuming unlimited supply of coins

Ex: $c_1 = 1, c_2 = 2, c_3 = 4, M = 6$.

Then $M = 6 \times c_1 = 4 \times c_1 + c_2 = 2 \times c_1 + 2 \times c_2$

$= 3 \times c_2 = 2 \times c_1 + c_3 = c_2 + c_3$

\[ \therefore 6 \text{ ways in total.} \]
DP Alg:

• Sub problems?
  
  Use less coin types, make $M'( \leq M)$ -cents change.

  $opt[i, j] = \# \text{ of distinct ways to make } j \text{-cents change}$
  
  using only coin types $c_1, \ldots, c_i$

  $opt[i, j] = \begin{cases} 
  0, & \text{if } i = 0 \\
  1, & \text{if } j = 0 \\
  opt[i-1, j], & \text{if } j < c_i \\
  opt[i-1, j] + opt[i, j-c_i], & \text{otherwise} 
  \end{cases}$

  $opt[n, M]$