COL106: Data Structures and Algorithms

Ragesh Jaiswal, IIT Delhi

Dynamic Programming

- Here is a *memoized* version of the recursive algorithm.

Algorithm

`LCS-mem`(S, n, T, m)
- If $(n = 0$ OR $m = 0)$ then return(0)
- If ($L[n, m]$ is known) then return($L[n, m]$)
- If $(S[n] = S[m])$
  - $length \leftarrow 1 +$ `LCS-mem`$(S, n-1, T, m-1)$
- If $(S[n] \neq T[m])$
  - $length \leftarrow \max\{$`LCS-mem`$(S, n, T, m-1)$,
    `LCS-mem`$(S, n-1, T, m)\}$
- $L[n, m] \leftarrow length$
- return($length$)

- What is the running time of the above algorithm? $O(nm)$

Problem

You are given n items with non-negative integer weights $w(i)$ and an integer W. You have to determine a subset S of $\{1, ..., n\}$ such that $\sum_{i \in S} w(i)$ is maximized subject to $\sum_{i \in S} w(i) \leq W$.

- Example: Let $(\{1, 2, 3, 5, 6, 7\}, 10)$ be the input instance.
- What is the optimal solution?

Problem

You are given n items with non-negative integer weights $w(i)$ and an integer W. You have to determine a subset S of $\{1, ..., n\}$ such that $\sum_{i \in S} w(i)$ is maximized subject to $\sum_{i \in S} w(i) \leq W$.

- Example: Let $([1, 2, 3, 5, 6, 7], 10)$ be the input instance.
- What is the optimal solution?$\{2, 3, 4\}$
  - Since $w(2) = 2, w(3) = 3, w(4) = 5$ and $w(2) + w(3) + w(4) = 10$.

Problem

You are given n items with non-negative integer weights $w(i)$ and an integer W. You have to determine a subset S of $\{1, ..., n\}$ such that $\sum_{i \in S} w(i)$ is maximized subject to $\sum_{i \in S} w(i) \leq W$.

- How do we define the subproblems for the Dynamic Program?

Problem

You are given n items with non-negative integer weights $w(i)$ and an integer W. You have to determine a subset S of $\{1, ..., n\}$ such that $\sum_{i \in S} w(i)$ is maximized subject to $\sum_{i \in S} w(i) \leq W$.

- How do we define the subproblems for the Dynamic Program?
- Let us try the following:
  - $M(i)$: The maximum weight that can be filled using items $\{1, ..., i\}$ subject to the sum being $\leq W$.
  - How do we define $M(i)$ in terms of $M(1), ..., M(i-1)$?

Problem

You are given n items with non-negative integer weights $w(i)$ and an integer W. You have to determine a subset S of $\{1, ..., n\}$ such that $\sum_{i \in S} w(i)$ is maximized subject to $\sum_{i \in S} w(i) \leq W$.

- How do we define the subproblems for the Dynamic Program?
- Let us try the following:
  - $M(i)$: The maximum weight that can be filled using items $\{1, ..., i\}$ subject to the sum being $\leq W$.
  - How do we define $M(i)$ in terms of $M(1), ..., M(i-1)$?
    - Case 1: i^{th} item is not in the optimal solution. Then $M(i) = M(i-1)$.
    - Case 2: i^{th} item is in the optimal solution. There is a problem here.

Problem

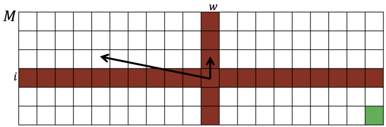
You are given n items with non-negative integer weights $w(i)$ and an integer W. You have to determine a subset S of $\{1, ..., n\}$ such that $\sum_{i \in S} w(i)$ is maximized subject to $\sum_{i \in S} w(i) \leq W$.

- How do we define the subproblems for the Dynamic Program?
- Let us try the following:
  - $M(i, w)$: The maximum weight that can be filled using items $\{1, ..., i\}$ subject to the sum being $\leq w$.
  - Recursive formulation:
    - Case 1: i^{th} item is not in the optimal solution. Then $M(i, w) = M(i - 1, w)$.
    - Case 2: i^{th} item is in the optimal solution. Then $M(i, w) = M(i - 1, w - w(i)) + w(i)$

Problem

You are given n items with non-negative integer weights $w(i)$ and an integer W. You have to determine a subset S of $\{1, ..., n\}$ such that $\sum_{i \in S} w(i)$ is maximized subject to $\sum_{i \in S} w(i) \leq W$.

- Dynamic Programming solution:
  - $M(i, w)$: The maximum weight that can be filled using items $\{1, ..., i\}$ subject to the sum being $\leq w$.
  - If $w(i) > w$, then $M(i, w) = M(i-1, w)$
  - If $w(i) \leq w$, then $M(i, w) = \max\{M(i-1, w), M(i-1, w - w(i)) + w(i)\}$
  - $\forall w \leq W$, $M(1, w) = w(1)$ if $w(1) \leq w$ and 0 otherwise.

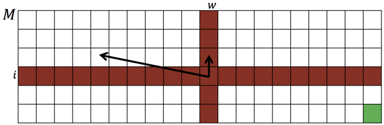


- What is the running time for filling the above table?

Problem

You are given n items with non-negative integer weights $w(i)$ and an integer W. You have to determine a subset S of $\{1, ..., n\}$ such that $\sum_{i \in S} w(i)$ is maximized subject to $\sum_{i \in S} w(i) \leq W$.

- Dynamic Programming solution:
  - $M(i, w)$: The maximum weight that can be filled using items $\{1, ..., i\}$ subject to the sum being $\leq w$.
  - If $w(i) > w$, then $M(i, w) = M(i-1, w)$
  - If $w(i) \leq w$, then $M(i, w) = \max\{M(i-1, w), M(i-1, w - w(i)) + w(i)\}$
  - $\forall w \leq W$, $M(1, w) = w(1)$ if $w(1) \leq w$ and 0 otherwise.



- What is the running time for filling the above table? $O(n \cdot W)$

End