Algorithm Lab

Week 6: 0/1 Knapsack Problem

Given a set of items $X = \{x_1, x_2, \dots, x_n\}$ and capacity C. We know the weight and value of each item, denoted by $W(x_i)$ and $V(x_i)$. The problem is to find a set $Y = \{y_1, y_2, \dots, y_m\} \subseteq X$ that $W(Y) = \sum_{y_i \in Y} W(y_i) \leq C$ and maximize $V(Y) = \sum_{y_i \in Y} V(y_i)$.

Instance: Capacity C and
a set of items $X = \{(W(x_1), V(x_1)), (W(x_2), V(x_2)), \dots, (W(x_n), V(x_n))\}$.

Result: A subset $Y \subseteq X$ that $W(Y) \leq C$ and maximize $V(Y)$.

Description

We can define the 0/1 knapsack problem as a function $f(C, X) = Y$.
If we have no capacity or have no item can take, we take nothing, respectively, $f(C, X) = \{\}$.
We can denote prefix subset by $X_i = \{x_1, x_2 \dots, x_i\} \subseteq X$.
Consider $f(C, X_i)$, if optimal solution is to take x_i, then $f(C, X_i) = \{x_i\} \cup f(C - W(x_i), X_{i-1})$;
If not to take x_i is better, than $f(C, X_i) = f(C, X_{i-1})$.

Thus, $f(C, X_i) = \begin{cases} \{\}, \ C \leq 0 \ or \ X_i = \{\} \\ \{x_i\} \cup f(C - W(x_i), X_{i-1}), \ V(\{x_i\} \cup f(C - W(x_i), X_{i-1})) > V(f(C, X_{i-1})) \\ f(C, X_{i-1}), \ V(\{x_i\} \cup f(C - W(x_i), X_{i-1})) \leq V(f(C, X_{i-1})) \end{cases}$

Questions

1. Analyze space and time complexity of a recursive implementation without cache.
2. Design a table to cache the answer of subproblems.
3. Analyze space and time complexity of implementation at Q2.
4. Please explain why the algorithm is a pseudo polynomial time algorithm (kind of exponential time algorithm), not a polynomial time algorithm.
5. Solve http://oj.csie.ndhu.edu.tw/problem/ALG05