
CE2101/CZ2101 ALGORITHM DESIGN AND ANALYSIS

Project 3: Dynamic Programming

We have a knapsack of capacity weight C (a positive integer) and n types of objects. Each object of the i th type has weight w_i and profit p_i (all w_i and all p_i are positive integers, $i = 0, 1, \dots, n-1$). There are unlimited supplies of each type of objects. Find the largest total profit of any set of the objects that fits in the knapsack.

Let $P(C)$ be the maximum profit that can be made by packing objects into the knapsack of capacity C .

- (1) Give a recursive definition of the function $P(C)$.
- (2) Draw the subproblem graph for $P(14)$ where n is 3 with the weights and profits given below.

	0	1	2
w_i	4	6	8
p_i	7	6	9

- (3) Give a dynamic programming algorithm to compute the maximum profit, given a knapsack of capacity C , n types of objects with weights w_i and profits p_i using the bottom up approach.
- (4) Code your algorithm in a programming language
 - a. show the running result of $P(14)$ with weights and profits given in (2).
 - b. Show the running result of $P(14)$ with weights and profits given below.You should show the program input and the output in (a) and (b).

	0	1	2
w_i	5	6	8
p_i	7	6	9