CS 6363.004: Algorithms: Spring 2018
Programming Project: Mon, Mar 5

Report: written/typed, and submitted in class/elearning. Code: upload on elearning.

Due: 10:00 AM, Thu, Apr 5.

Programs can be written in Java, C++, or, Python. Preferred language is Java, and some starter code in Java will be provided. In addition, write or type the algorithm and submit proof of correctness and running time analysis.

Ver 1.0: Initial description (Mar 5, 10:30 AM).

Ver 1.1: Added explanation that "profit of p_i" is per unit of j_i sold. Error in sample output corrected. Optimal profit is 98 and not 18, as given earlier.

Scenario: A jeweller has G units of gold, and has signed contracts to make n jewelry items, $j_1..j_n$. The following are the conditions, $i = 1..n$:

- Item j_i requires w_i units of gold to make, and generates a profit of p_i per unit of j_i. The agreement is to supply a minimum quantity of n_i, and a maximum quantity of x_i.
- Additional quantities of j_i beyond x_i cannot be sold and do not generate any profit.
- Not meeting the minimum quantities promised in the contracts triggers penalty fines. For j_i, if the quantity supplied (q_i) is below n_i, then the jeweller has to pay a fine of f_i for each unit below the minimum, with a cap of total fine of c_i, i.e., when $q_i < n_i$, fine for j_i is calculated as $\min(f_i * (n_i - q_i), c_i)$.

Write a 1 recurrence for the problem of optimizing the net profit (total profit generated by items supplied minus the fines paid), and the number of different optimal solutions that are possible. 2 Prove the correctness of these recurrences. 3 Write a dynamic program corresponding to the recurrences. Write a program that implements the DP.

Input specification: If no command line parameter is supplied, or the first parameter is "−", the input is read from the console. Otherwise, the first parameter is used as the name of a file that has the input. The first two numbers in the input are G and n. The next n lines of the input have 7 integers each: $i, w_i, p_i, n_i, x_i, f_i, c_i$. If there are additional lines in the input, ignore them.

Output specification: Normally the program prints just 2 numbers in the output: maximum net profit, and the number of optimal solutions. If there is a second command line parameter, whose value is greater than 0, then the list of distinct optimal solutions are enumerated, one per line. Do not print any other captions or unnecessary output.

Sample input and output:

G N

i, wi, pi, ni, xi, fi, ci

```
20 2
1 2 10 8 10 2 2
2 2 10 8 10 2 4
# Output:
```

maximum net profit, and the number of optimal solutions

```
98 3
# List of optimal solutions:
2 8
1 9
0 10
```

Explanation: Item 1 has $w_1 = 2, p_1 = 10$, minimum quantity $n_1 = 8$, maximum quantity $x_1 = 10$, penalty fine of 2 per item below n_i, capped at 2. Item 2 has the same parameters, except the fine is capped at 4. The available gold of 20 units can be used to make only 10 items, leaving a deficit of 6 items (in meeting the minimums). The optimal solution chooses to have the deficits on item 1 because of the cap on maximum fine.