FIT2004 S2/2017: Assessment questions for week 6

THIS PRAC IS ASSESSED! (5 Marks)

DEADLINE: Sunday, 27-Aug-2017 23:55:00 AEST
CLASS: You will be interviewed during your lab by your demonstrator who will ask you a series of questions to assess your understanding of this exercise, and gauge how you implemented it. It is required that you implement this exercise strictly using **Python programming language**. Practical work is marked on the time and space complexity of your program **and also on your understanding of the program**. A *perfect* program with zero understanding implies you will get **zero** marks! "Forgetting" is not an acceptable explanation for lack of understanding. Demonstrators are not obliged to mark programs that do not run or that crash.
After/before your demonstrators have interviewed you, you are expected to work towards the programming competition the details of which will be released soon.
SUBMISSION REQUIREMENT: You will need to submit a zipped file containing your Python program (named ebay.py) as well as a PDF file briefly describing your solution and its space and time complexity. The PDF file must give an outline of your solution (e.g., a high level idea of how did you solve it) and the **worst-case** space and time complexity of your solution. Penalties will be applied if you fail to submit the PDF file. The zipped file is to be submitted on Moodle before the deadline.
Important: The assignments will be checked for plagiarism using an advanced plagiarism detector and the students will be interviewed by tutors to demonstrate the understanding of their code. Last year, many students were detected by the plagiarism detector and almost all got zero mark for the assignment and, as a result, many failed the unit. "Helping" others is NOT okay. Please do not share your solutions with others. If someone asks you for help, ask them to visit us during consultation hours for help.

Alice in ebayland

The last few days have been quite busy as you had been working hard on the Scrabble Helper for Alice. However, you are happy that you were able to finish the program before her rematch with Cindy. You are hoping that your hard work pays off and Alice wins the match. While you are thinking about her rematch, you hear a knock on the door. It is Alice – very excited Alice.

Alice: "Yayyyyyy!!! I won by a huge margin. Thank you, thank you, thank you! Thank you so much for all your hard work"

A part of you wanted to tell her how hard you had worked on the program but another part wanted to brag about your algorithmic skills. You chose the latter: "No problem at all! It took me hardly 30 minutes. Don't embarrass me, it's not a big deal! Let's change the topic! What's new?"

Alice: "Well, I am starting an online business on ebay. I have done research about around 50 products and have obtained the cost per item and expected profit for each product. I need to choose some of these products to sell on ebay. However, ... "

You can see where it is going and you're hoping that she does not ask you to write another program. However, you have to pay for bragging about your algorithmic skills.

Alice continues: "However, I don't know what are the best products to choose given some constraints. Since you are very good at algorithms and data structures, I am sure you will be able to help me again."

You: "hmmm... okay, give me more details"

Alice: "Well, I am a new seller on ebay and ebay applies a restriction on new ebay sellers which reads something like ***You can sell up to 30 items or up to 1,000 AUD per month, whichever comes first.*** In other words, there are two types of restrictions on new sellers: 1) *item limit* is a limit on the number of items I can sell per month (30 items per month in the above example); 2) *price limit* is the limit on the total price of items I can sell per month (1,000 AUD per month in the above example). Given the restrictions, I want to select the products based on their price per item and profit per item such that the total profit in a month is maximized."

You: "This sounds like a Dynamic Programming problem. Tell me more details".

Alice is very excited to hear that you can help her. She explains the detailed requirements which are formally described below.

Input

The input file `products.txt` (available on Moodle) contains information about 52 products. Below are the first five lines from the input file.

```
0:Cool Blue Marella Jug:33:15
1:Weight Loss Pack:55:16
2:Natural Black Vogue Lashes:10:6
3:Paris Age Perfect Intense Nutrition Serum:45:22
4:Age Perfect Night Cream:36:12
```

Values in each line are separated by colons ":". The first value is the product ID (0 to 51), the second value is the name of the product, the third value is an integer corresponding to the price (in AUD) per item for the product and the fourth is an integer corresponding to the profit (in AUD) per item for the product. For example, product ID for "Cool Blue Marella Jug" is 0, price for a single "Cool Blue Marella Jug" is 33 AUD and profit for selling one jug is 15 AUD.

Task 1: Maximize profit when there is no item limit

Alice contacted ebay and convinced them to waive the *item limit* for the first month. In other words, for the first month, she is allowed to sell as many items as she wants as long as the total price of the sold items is less than or equal to the *price limit*. Therefore, she wants to select the products such that their total price is at most equal to the price limit and the total profit is maximized. You can assume that there is an infinite supply for every product, e.g., she can sell as many items of any given product as she wants as long as the total price is less than the price limit.

You will need to write a function named `MaximizeProfit1` which determines the products to maximize the profit for a given price limit. Below are some sample outputs for the data in `products.txt` and different values of price limit.

```
Enter the price limit: 1500
165  X  ['Lint Pet Roller', 9, 6]
1  X  ['Lint Roller Refill', 8, 5]
1  X  ['Gluten Free Mixed Seeds', 7, 4]
Total Price of items sold:  1500
Total Items sold:  167
Total Profit:  999
```

The above example shows that if the price limit is $1,500$ AUD and there is no item limit, the profit can be maximized by selling 165 Lint Pet Rollers, 1 Lint Roller Refill and 1 Gluten Free Mixed Seeds. The total price of items sold in this case is 1500, the total items sold is 167 and the total profit Alice would earn is 999 AUD. The values next to products are the price per item and profit per item, e.g., the price per item for Lint Pet Roller is 9 AUD and profit per Lint Pet Roller is 6 AUD.

Below are some other sample outputs for different price limits.

```
Enter the price limit: 50
4  X  ['Lint Pet Roller', 9, 6]
2  X  ['Gluten Free Mixed Seeds', 7, 4]
Total Price of items sold:  50
Total Items sold:  6
Total Profit:  32
```

```
Enter the price limit: 750
83  X  ['Lint Pet Roller', 9, 6]
Total Price of items sold:  747
Total Items sold:  83
Total Profit:  498
```

Note that in the above example the price limit is 750 AUD and the total price for items sold is 747 AUD, i.e., the total price for the items sold by the optimal strategy may be smaller than the price limit.

```
Enter the price limit: 3300
365  X  ['Lint Pet Roller', 9, 6]
1  X  ['Lint Roller Refill', 8, 5]
1  X  ['Gluten Free Mixed Seeds', 7, 4]
Total Price of items sold:  3300
Total Items sold:  367
Total Profit:  2199
```

Complexity Requirements: Let N be the total number of products in `products.txt` and P be the price limit. Your algorithm must print the results in $O(PN)$ worst-case time complexity and must have $O(N + P)$ worst-case space complexity.

Task 2: Maximize profit with both item limit and price limit

In this task, Alice wants to find the strategy to maximize profit given both the item limit and price limit. Let M be the item limit and P be the price limit, the goal is to sell at most M items with total price at most P such that the profit is maximized.

In this task, you will write a function called `MaximizeProfit2` which determines the products that maximize the profit for a given item limit and a given price limit. Below are some sample outputs for the data in `products.txt` and different values for item limit and price limit.

```
Enter the price limit: 1500
Enter the item limit: 24
4  X  ['Revitalift Filler Day Cream', 49, 26]
1  X  ['Paris Age Perfect Cell Renewal Day Cream', 50, 27]
19  X  ['Pro Clinical White C250 Toothbrush', 66, 32]
Total Price of items sold:  1500
Total Items sold:  24
Total Profit:  739
```

In the above example, the price limit is 1500 AUD and item limit is 24 items. The profit is maximized by selling 4 Revitalift Filler Day Creams, 1 Paris Age Perfect Cell Renewal Day Creams and 19 Pro Clinical White C250 Toothbrushes. Note that the total number of items sold is at most 24 which satisfies the item limit and the total price of items sold is 1500 which satisfies the price limit. The total profit in this case is 739 AUD. Below are some more examples.

```
Enter the price limit: 50
Enter the item limit: 4
1  X  ['Liquid Cut & Polish', 18, 10]
2  X  ['Lint Pet Roller', 9, 6]
1  X  ['Chckn&trk All Breed D/f', 14, 8]
Total Price of items sold:  50
Total Items sold:  4
Total Profit:  30
```

```
Enter the price limit: 750
Enter the item limit: 20
9  X  ['Age Perfect Intense Nutrition Night Cream', 40, 22]
1  X  ['Paris Age Perfect Cell Renewal Day Cream', 50, 27]
10  X  ['Intense Nutrition Repairing Day Cream', 34, 19]
Total Price of items sold:  750
Total Items sold:  20
Total Profit:  415
```

```
Enter the price limit: 3300
Enter the item limit: 30
10  X  ['Pro Clinical White C250 Toothbrush', 66, 32]
20  X  ['Prepaid HTC Desire 320', 129, 37]
Total Price of items sold:  3240
Total Items sold:  30
Total Profit:  1060
```

Complexity Requirements: An exact requirement on time complexity is not being given because this would give a hint on how to solve the problem. Your algorithm must not have exponential time or space complexity. You must use a dynamic programming strategy to solve the problem. You can assume that the item limit will be at most 35 and price limit will be at most 3500. If you have implemented a DP solution correctly, your algorithm will be able to return the results within a few seconds.

Submission Requirements

Your final program should ask the user to enter a price limit and an item limit. It should then call the function created in Task 1 to print the strategy to maximize profit where only price limit is applied and then it should call the function created in Task 2 to print the strategy to maximize the profit where both price and item limits are applied. Below is a sample output – your final program must follow this output format.

```
Enter the price limit: 1500
Enter the item limit: 24

Strategy for price limit only
******************************
165  X  ['Lint Pet Roller', 9, 6]
1  X  ['Lint Roller Refill', 8, 5]
1  X  ['Gluten Free Mixed Seeds', 7, 4]
Total Price of items sold:  1500
Total Items sold:  167
Total Profit:  999

Strategy for both item and price limit
**************************************
4  X  ['Revitalift Filler Day Cream', 49, 26]
1  X  ['Paris Age Perfect Cell Renewal Day Cream', 50, 27]
19  X  ['Pro Clinical White C250 Toothbrush', 66, 32]
Total Price of items sold:  1500
Total Items sold:  24
Total Profit:  739
```

Important: There may be multiple ways to choose the products to maximize the profit. You can print any set of products that gives the maximum profit. In other words, your output may contain a different set of products but as long as the maximum profit is the same as the sample output and the limits are satisfied, your answer will be considered correct. Also, the order of products in the output is not important.

Things to note

Your programs will be tested on a different `products.txt` file which may have different number of products, prices and profits. While a greedy approach may work on some test cases in the given file, it may not always work. You will need to write an algorithm that produces correct results for any given input. Even if your program gives correct output for the test cases given

above, it does not necessarily guarantee that your algorithm is correct. It is your responsibility to ensure the correctness of your algorithm. You can assume that the IDs of the products in the input file are sequential and start from 0, e.g., in the example, IDs range from 0 to 51.

Important: It is strongly recommended that you start working on the assignment as early as possible. It may take longer than you expect. Also, note that I may not check/reply emails during the weekend. Therefore, you should aim to finish your assignment by Friday evening.

```
-=o0o=-
  END
-=o0o=-
```