ENSF 594

Assignment 2

Due Time: 7/ 26/ 2020

This assignment has three different sections about greedy algorithms.

What to upload:

1- A report including the screen shot of your result (PDF format)
2- Your code.

1 Money Change

Problem Introduction

In this problem, you will design and implement an elementary greedy algorithm used by cashiers all over the world millions of times per day.

Problem Description

Task. The goal in this problem is to find the minimum number of coins needed to change the input value (an integer) into coins with denominations 1, 5, and 10.

Input Format. The input consists of a single integer m.

Constraints. $1 \leq m \leq 10^3$.

Output Format. Output the minimum number of coins with denominations 1, 5, 10 that changes m.

Sample 1.

Input:

```
2
```

Output:

```
2
```

$2 = 1 + 1$.

Sample 2.

Input:

```
28
```

Output:

```
6
```

$28 = 10 + 10 + 5 + 1 + 1 + 1$.

2 Maximum Value of the Loot

Problem Introduction

A thief finds much more loot than his bag can fit. Help him to find the most valuable combination of items assuming that any fraction of a loot item can be put into his bag.

Problem Description

Task. The goal of this code problem is to implement an algorithm for the fractional knapsack problem.

Input Format. The first line of the input contains the number n of items and the capacity W of a knapsack. The next n lines define the values and weights of the items. The i-th line contains integers v_i and w_i—the value and the weight of i-th item, respectively.

Constraints. $1 \leq n \leq 10^3$, $0 \leq W \leq 2 \cdot 10^6$; $0 \leq v_i \leq 2 \cdot 10^6$, $0 < w_i \leq 2 \cdot 10^6$ for all $1 \leq i \leq n$. All the numbers are integers.

Output Format. Output the maximal value of fractions of items that fit into the knapsack. The absolute value of the difference between the answer of your program and the optimal value should be at most 10^{-3}. To ensure this, output your answer with at least four digits after the decimal point (otherwise your answer, while being computed correctly, can turn out to be wrong because of rounding issues).

Sample 1.

Input:

```
3 50
60 20
100 50
120 30
```

Output:

```
180.0000
```

To achieve the value 180, we take the first item and the third item into the bag.

Sample 2.

Input:

```
1 10
500 30
```

Output:

```
166.6667
```

Here, we just take one third of the only available item.

3 Maximum Advertisement Revenue

Problem Introduction

You have n ads to place on a popular Internet page. For each ad, you know how much is the advertiser willing to pay for one click on this ad. You have set up n slots on your page and estimated the expected number of clicks per day for each slot. Now, your goal is to distribute the ads among the slots to maximize the total revenue.

Problem Description

Task. Given two sequences $a_1, a_2, \ldots, a_n$ (a_i is the profit per click of the i-th ad) and $b_1, b_2, \ldots, b_n$ (b_i is the average number of clicks per day of the i-th slot), we need to partition them into n pairs (a_i, b_j) such that the sum of their products is maximized.

Input Format. The first line contains an integer n, the second one contains a sequence of integers $a_1, a_2, \ldots, a_n$, the third one contains a sequence of integers $b_1, b_2, \ldots, b_n$.

Constraints. $1 \le n \le 10^3$; $-10^5 \le a_i, b_i \le 10^5$ for all $1 \le i \le n$.

Output Format. Output the maximum value of $\sum_{i=1}^{n} a_i c_i$, where $c_1, c_2, \ldots, c_n$ is a permutation of $b_1, b_2, \ldots, b_n$.

Sample 1.

Input:

```
1
23
39
```

Output:

```
897
```

$897 = 23 \cdot 39$.

Sample 2.

Input:

```
3
1 3 -5
-2 4 1
```

Output:

```
23
```

$23 = 3 \cdot 4 + 1 \cdot 1 + (-5) \cdot (-2)$.