

20

Task Manager

11 points

Introduction

Adrià is in charge of evaluating the problems proposed for this year's edition of CodeWars. Most of these problems are designed by experts inside HP, and when they send them to be supervised, they want to know if the problem submitted is valid or if it needs changes. These experts are very busy and need to get an answer as soon as possible.

Luckily for Adrià, he programmed a few weeks ago an AI that estimates the time it would take him to supervise a problem. However, Adrià is a bit lazy and doesn't want to figure out, every day, the order of the tasks.

Could you give a hand to him? He needs to find a way to sort the problems so that the total time waited by all experts is the minimum.

HINTS:

- Keep in mind Adrià is not a superhero, he can only supervise **one problem at a time**.
- In case two problems have the same priority, sort in the order they were read from input.

Input

The input will consist of two lines

The first line contains the names of the problems

The second line contains integers with the time needed to evaluate the problem. Each number corresponds to a unique problem.

Realize that both lines must have the same number of elements.

Output

The output should be a list of problems sorted by priority with the time they had to wait and the total time waited.

Example 1

Input

Task1 Task2

3 2

Output

Task2 wait time: 2

Task1 wait time: 5

Total time waited: 7

Example 2

Input

Prob1 Prob2 Prob3 Prob4

3 1 1 2

Output

Prob2 wait time: 1

Prob3 wait time: 2

Prob4 wait time: 4

Prob1 wait time: 7

Total time waited: 14

Example 3

Input

```
A B C D E F G H I J K L  
9 2 5 3 1 1 4 10 3 5 4 3
```

Output

```
E wait time: 1  
F wait time: 2  
B wait time: 4  
D wait time: 7  
I wait time: 10  
L wait time: 13  
G wait time: 17  
K wait time: 21  
C wait time: 26  
J wait time: 31  
A wait time: 40  
H wait time: 50  
Total time waited: 222
```