

Game

Objective

The objective of this problem is to test the understanding of Object-Oriented Programming (OOP) concepts, in particular, on inheritance.

Problem Description

You are playing a game named “Square Game”, a simplified version of snake-and-ladder board game. In this game, you have M squares numbered from 0 to $M - 1$.

You begin in square 0 and play by throwing dice. Each turn you move by the number shown on the dice, and land in a square. There are two types of squares: *Normal Square* and *Special Square*. Each type has a “value” attribute which can be positive or negative, and Special Square has one more attribute “multiplier”.

When you land on a Normal Square, you get points according to the “value” of the square. This “value” will be added to your points. When you land on a Special Square, you multiply your current score with the “multiplier” first, then add the “value”.

Given the dice values, you are to simulate the game and print the final score. If you throw the dice and get past the last square (square $M - 1$), continue on to square 0.

Input

The input starts with the number M ($1 \leq M \leq 1000$) representing the number of squares. This is followed by M lines where each line contains two or three numbers. The first number of the line indicates the square type. If it is 0 then it is a normal square and the value will follow. If it is 1 then it is a special square and the value and multiplier will follow. The last line in the input contains a sequence of numbers (terminated by end-of-file) representing the values shown on the dice in each turn.

Output

The output will be in format: “Final score: <score>” where <score> represents your final score after playing the game. Print the final score correct to two decimal places.

Sample Input

```
5
0 5.25
0 -3.14
1 2.17 1.50
0 5.39
1 -3.32 2.00
3 3 1 2
```

Sample Output

```
Final score: 7.77
```

Explanation

After the first turn we are on square 3, hence we add 5.39 to our points. (Total: 5.39)

After the second turn we are on square 1 $((3+3)\%5)$, hence we add -3.14 to our points (Total: 2.25)

After the third turn we are on square 2 $(1+1)$, hence we multiply by 1.5 and add 2.17 (Total: 5.545)

After the fourth turn we are on square 4 $(2+2)$, hence we multiply by 2 and add -3.32 (Total: 7.77)

Note

1. Please use this problem as a platform to practice OOP principles.
2. The main Java class must be called **Game**, and be in the source file **Game.java**.