Calculate Percentile Score

You have just received your last semester result. Your professor has asked you to write a program to compute your percentile score.

The percentile score can be calculated as follows:

If you have scored S marks, then find all students whose score is less than S. Let's assume there are X students whose score is less than S and there are N students in total. Given this information, your percentile score would be: X / N * 100

Inputs:

1. Total Students - *Integer*
2. Marks obtained by each student - *Array of integers of length N*
3. Your score - *Integer*

Output:

1. Your percentile score - *float*

Note: Your answer should be correct up to 2 decimal places.

Example:

Inputs:

1. 7
2. [800, 300, 950, 760, 680, 490, 640]
3. 760

Output:

57.14

Explanation:

1. There are total 7 students
2. Your score is 760, and there are 4 students, whose score is less than 760, i.e. 300, 680, 490, 640
3. Percentile = 4 / 7 * 100 = 57.14
   Hint: Use floating point operation

Students Screener

Your college is requesting applications for scholarships and wants to pick students based on their score in the last exam. The college needs your help to write a program to find eligible students. The program will receive lists of students, marks obtained, total marks, exam result (Passed/Failed) as input, and eligibility criteria in terms of percentile. The program should return a list of students who are eligible for the scholarship.

Assumptions:

1. There will be an utmost 1000 students. Total Marks, Obtained marks, and required percentile will be in the whole integer number.
2. Each student's score will be different from any other student, i.e. there will not be multiple students with the same score.

Hint: Since there won't be multiple students with same score, the percentile score of a student can be computed as follows:

Given there are total N students, find rank of each student and then use below formula to compute the percentile score of a student, whose rank is R

Percentile = (N - R) / N * 100

Inputs:

1. Total marks - *integer*
2. List of Student (name) - *Array of strings of length N*
3. Marks obtained by each student - *Array of integers of length N*
4. Exam result for each student - *Array of strings (Passed/Failed) of length N*
5. Required Percentile to be Eligible for scholarship - *Integer*

Output:

1. Names of students, comma separated, who have passed the exam and who has scored X percentile or above, in descending order of their score

Example:

Inputs:

4. 1000
5. [“Kartik”, “Devang”, “Pari”, “Ketan”, “Sheetal”, “Darshana”, “Mohan”]

6. [800, 300, 750, 760, 680, 790, 640]
7. [“Passed”, “Failed”, “Passed”, “Failed”, “Passed”, “Passed”, “Passed”]
8. 50

Output:

“Kartik,Darshana”

Explanation:

1. Kartik has scored highest, so his rank is 1. Percentile = (7-1) / 7 * 100 = 85.71
2. Devang has scored lowest, so his rank is 7. Percentile = (7-7) / 7 * 100 = 0
3. Pari’s rank is 4. Percentile = 42.86
4. Ketan’s rank is 3. Percentile = 57.14
5. Sheetal’s rank is 5. Percentile = 28.57
6. Darshana’s rank is 2. Percentile = 71.43
7. Mohan’s rank is 6. Percentile = 14.29

From the above list, the percentile score of Kartik, Ketan and Darshana is greater or equal to 50. However, Ketan did not pass the exam (as he might have failed in one or more subjects).

So, only Kartik and Darshana are qualified, and if we sort them based on their scores (in descending order) and return their names using comma separated string, it will be: **“**Kartik,Darshana”

Trivia: Even though Kartik is topper, his percentile score is 85.71, because there are only 7 students. If there are 10,000 or more students, the percentile score of the first ranker will be greater or equal to 99.99

Guess the Number

Your niece has been taught a fun math game using interesting math properties. The game can be played as - assume a number X and apply a series of mathematical operations to arrive at the final number.

Given this final number N and the series of operations she asked you to do as part of the game, find the original number A so that if you apply all the operations in the same order, you will arrive at the final number N.

Inputs:

1. Final number - *Integer*
2. Series of operations - *Array of strings*
    a. Each element of the array will represent an operation.

    An operation will in format of: X OPERATOR VALUE

    OPERATOR will be: +, -, *, /, %, ^
    VALUE will be: A non-negative integer number, i.e. an integer number greater or equal to zero

    See below examples for the details:

    X + 15 (Add 15 to the number)
    X - 0 (Subtract 0 from the number)
    X * 82 (Multiply the number with 82)
    X / 14 (Divide the number by 14)
    X ^ 6 (X raised to power 6)
    X % 8 (Divide X by 8 and return reminder, i.e. modulo operation)

    There will be a single space before and after the operator.

Output:

A positive integer number A on which, if all operations are applied in the same order, we will arrive at the Final Number, given in Input-1.

Notes:

1. Some of the mathematical operations could be invalid. For example, divide by zero. If you find any of such operation, return -1
   See Example-2 for details

2. Some of these operations result in multiple answers for original number A.
   See Example-3 for details. If you find such operation, return -2

3. You can use library functions to do string parsing

4. You can use library functions to do certain mathematical operations, e.g. power or exponent

Examples:

Example-1:

Inputs:

1. 2500
2. [“X + 10”, “X - 5”, “X * 5”, X ^ 2”]

Output: 5

Explanation:
If all operations specified in the Input 2 are applied on 5, we’ll arrive at the number equal to 2500 (Input 1). i.e.

5 + 10 = 15
15 - 5 = 10
10 * 5 = 50
50 ^ 2 = 50 * 50 = 2500

Example-2:

Inputs:

1. 1000
2. [“X * 5”, “X / 0”, “X ^ 3”]

Output: -1

Explanation:
2nd operation (X / 0) is an invalid operation.

Example-3:

Inputs:

1. 10
2. [“X * 5”, “X * 0”, “X + 10”]

Output: -2

Explanation:
Here, 2nd operation (X * 0), is the operation, which results in multiple answers. Because irrespective of the initial number, your final answer will always be 10.

For example,

Let’s assume that the initial number is 3, and if you apply all these operations,
3 * 5 = 15
15 * 0 = 0
0 + 10 = 10

Let’s assume that the initial number is 8, and if you apply all these operations,
8 * 5 = 40
40 * 0 = 0
0 + 10 = 10

That is, it is impossible to find a unique answer to this problem and hence return -2.

Hint:

1. Multiplication (*), Exponent (^) and Modulo (%) are such operations wherein if they are used with some non-negative integers, it’s impossible to find the unique original number, as explained above.

Example-4:

Inputs:

1. 617283948
2. [“X + 5”, “X - 0”, “X + 1”, “X / 2”, “X ^ 1”]

Output: 1234567890
Note: Your answer can be a large number, but it will fit into a 32 bit integer

Explanation:

1234567890 + 5 **=** 1234567895
1234567895 - 0 = 1234567895

1234567895 + 1 = 1234567896
1234567896 / 2 = 617283948
617283948 ^ 1 = 617283948