Indian Statistical Institute
M.Tech.(CS) — First Year
Programming Test 2024
Date: 26.07.2024 Duration: 16:30–18:30

INSTRUCTIONS

1. Unless otherwise specified, your program should take input from stdin and print its output for stdout.
2. Please make sure that your programs adhere strictly to the specified input and output format.
3. To pass this test, you must submit solutions that pass **ALL** test-cases for at least 2 out of the following 3 problems.

1. Consider a reservoir, fitted with a set of taps and drains. Write a program that takes as input details about the taps and drains attached to the reservoir, and prints the correct output from among the following options.

   (A) If all taps and drains are kept open, *a partially empty reservoir never fills up.*
   (B) If all taps and drains are kept open, *an initially empty reservoir fills up in* _____ *minutes.*
   (C) If all taps and drains are kept open. *a partially full reservoir never becomes empty.*
   (D) If all taps and drains are kept open, *an initially full reservoir becomes empty in* _____ *minutes.*

   Input format: The input will consist of

   - 2 integers, M and N, respectively denoting the number of taps and drains attached to the reservoir, followed by
   - M floating point numbers, say t_1, t_2, …, t_M, with t_i denoting the time (in minutes) that it takes to fill the reservoir if it is initially empty, all drains are stopped, and only the i-th tap is opened ($1 \leq i \leq M$); followed by
   - N floating point numbers, say d_1, d_2, …, d_N, with d_j denoting the time (in minutes) that it takes to drain the reservoir if it is initially full, all taps are shut off, and only the j-th drain is opened ($1 \leq j \leq N$).

   Output format: Your program should print the letter(s) (A, B, C and/or D) corresponding to the correct option(s) for the given data, in alphabetic order. Each letter should appear on a separate line. For options (B) and (D), your program should also print the appropriate number of minutes as a floating point number, correct to 2 decimal places, in place of the blank. This floating point number should be printed to the right of the letter, separated by a single space character.

   Sample input 1:

   ```
   2 3
   1 1
   1 1 1
   ```

   Sample output 1:

   ```
   A
   D 1.00
   ```

   Explanation: There are two taps and three drains, all of which have the same flow rate. Two taps and two drains "cancel each other out", so we are effectively left with a reservoir with one drain that drains the full reservoir in 1 minute.

Sample input 2:

```
2 3
5 50
10 20 30
```

Sample output 2:

```
B 27.27
C
```

Explanation: In one minute, the two taps together fill $\frac{1}{5} + \frac{1}{50} = \frac{11}{50}$ of the reservoir, while the three drains together empty $\frac{1}{10} + \frac{1}{20} + \frac{1}{30} = \frac{11}{60}$ of the reservoir. Since the combined capacity of the taps is more than that of the drains, options (B) and (C) are correct.

2. Consider a point that starts from the origin, and moves East, West, North or South for M steps. Write a program that takes the directions of the M steps, and computes the final distance of the point from the origin. (Assume that in each step, the point travels unit distance.)

   Input format: A sequence of letters from the set $\{E, W, N, S\}$ denoting the direction of movement of the point at each step. The length of the sequence will **NOT** be given to you in advance. The first character not belonging to the set will mark the end of the input.

   Output format: Your program should print the distance from the final position of the point to the origin, as a floating point number.

   Sample input 1: `EWNS`

   Sample output 1: `0.000000`

   Sample input 1: `WSSW`

   Sample output 1: `2.828283`

3. A sequence $\{s_1, s_2, \ldots, s_m\}$ of $m > 1$ strings is said to satisfy property $\mathcal{P}$ if

   - all the strings s_1, s_2, …, s_m are of the same length; and
   - for any i $(1 \leq i \leq m-1)$, the strings s_i and s_{i+1} differ in *at most* two positions.

   Write a program that takes a sequence of strings as input, and determines whether the strings satisfy property $\mathcal{P}$ or not.

   Input format: The input will consist of a positive integer m, followed by m strings. Each string will appear on a line by itself, and will consist of at most 63 lowercase letters.

   Output format: Your program should print either YES or NO, depending on whether the given sequence satisfies property $\mathcal{P}$ or not.

Sample input 1:

```
4
cost
boat
boot
foot
```

Explanation: From `cost` to `boat`, 2 letters have changed; for each subsequent pair, exactly 1 letter has changed.

Sample output 1: `YES`

Sample input 2:

```
4
abcdefgh
abcdefgh
abxdefgy
abxdefgyz
```

Sample output 2: `NO`

Explanation: From s_1 to s_2, there is no change; for the next pair, 2 letters have changed, but for the last pair, the length has changed.