

Indian Statistical Institute
Semester-I 2023–2024
M.Tech.(CS) - First Year
Subject: Introduction to Programming
Backpaper Examination
Total: 60 marks Duration: 3 + 1 hrs.

INSTRUCTIONS

1. You may consult or use slides / programs provided to you as course material, or programs that you have written yourself as part of classwork / homework for this course (or the Computing Laboratory course), but please **do not** consult or use material from other Internet sources, your classmates, or anyone else.
2. Please make sure that your programs adhere strictly to the specified input and output format. **Your program may not pass the test cases provided, if your program violates the input and output requirements.**
3. Submissions from different students having significant match will be **debarred from evaluation.**

NOTE: Unless otherwise specified, all programs should take the required inputs from stdin, and print the desired outputs to stdout.

Q1. (15 marks)

Consider the following type of problem that is typically assigned to students in primary school.

*Person **A** can tile a floor of area **a** in n_a days. Person **B** can tile a floor of area **b** in n_b days.
In how many days can a floor of area **c** be tiled if **A** and **B** work together?*

You have to write a program to solve a generalised version of the above problem.

Input format: The first line of the input will consist of a single integer N . This will be followed by N lines, each containing 3 space-separated fields: a person's name consisting only of letters (A–Z, a–z), a floating point number corresponding to an area, and a second floating point number specifying the number of days that the named person will need to tile a floor of the given area. The person's name will be at most 16 characters long.

The above input ($N + 1$ lines in all) will be followed by an integer M and M more “query” lines. Each query line will have the following format: it will start with an integer (k , say), followed by k names (strings of the same format as above), followed by a floating point number corresponding to an area.

Sample input 0:

```

3                                ← # of persons
Prasanta 20.0 3.0              ← Prasanta can tile an area of 20 units in 3 days
Satyen 10.5 2                  ← Satyen can tile an area of 10.5 units in 2 days
Meghnad 15 1                   ← Meghnad can tile an area of 15 units in 1 day
2                               ← Query lines start from the next line
1 Prasanta 40                  ← In how many days can Prasanta tile an area of 40 units?
2 Meghnad Prasanta 100        ← In how many days can Meghnad and Prasanta together
                               tile an area of 100 units?

```

Output format: For each of the M query lines, your program should print the answer to the corresponding question as a single floating point number correct to 4 decimal places.

Sample output 0:

```

6.0000 ← Prasanta needs 6.0000 days to tile an area of 40 units.
4.6154 ← Meghnad and Prasanta need about 4.6154 days to tile an area of 100
        units when they work together.

```

Q2. (20 marks)

You are given a text file containing batting statistics for a group of cricketers. Write a program to find the names of cricketers who have scored at least two centuries but have also scored zero runs (a ‘duck’) in at least one innings. Your program should output these players’ names, the names of the countries they represent, and their batting averages. The output should be sorted in alphabetical order of the players’ names.

Input format: Your program should take a single command-line argument which specifies the name of the text file containing the batting data.

The file will consist of one line per cricketer. The number of lines in the file will not be given to you in advance. Each line will contain

- a player’s name (a single string consisting only of letters);
- the country s/he represents (same format as above);
- the total number of innings played by the cricketer; and
- the runs scored by the player in each innings.

The players’ names and country names will be less than 32 characters long.

Output format: Your program should print one line per selected player. Each line should contain 3 space-separated fields: the player’s name, country and batting average (correct to 2 decimal places).

Sample input 0: (content of input file)

```

Vinoos India 5 50 60 70 0 90
Virat India 11 101 145 0 15 26 90 44 83 61 0 4
Donald Australia 8 177 96 59 141 87 123 157 0

```

Sample output 0:

Donald Australia 105.00

Virat India 51.73

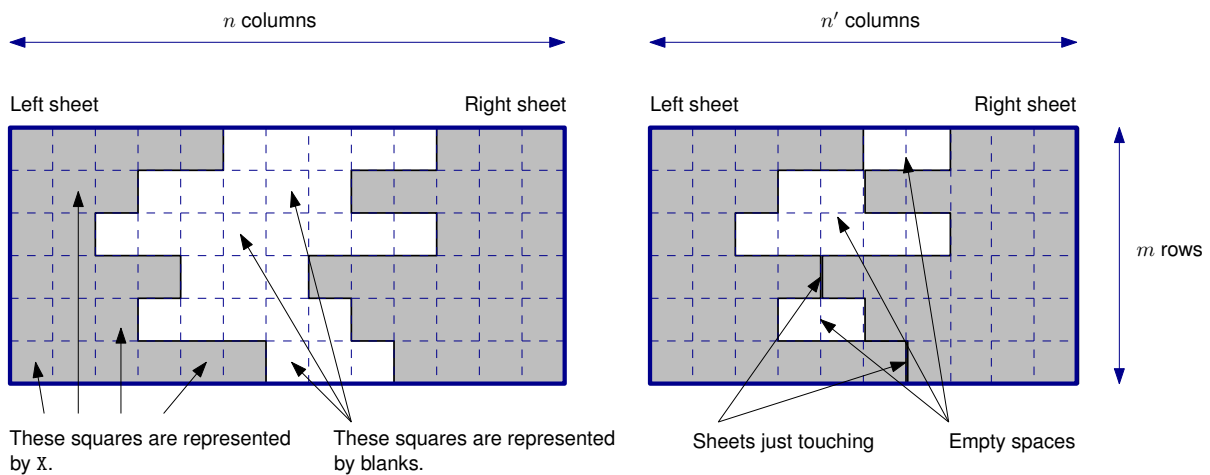
Q3. (25 marks)

You are given two sheets of paper. One edge of each sheet is cut in the shape of a histogram. The two sheets are then placed on a table with their top and bottom edges aligned, and the rough edges facing each other, as shown in Figure 3(a) below. The sheets are then moved towards each other, until they just touch (see Figure 3(b)). You have to write a program that determines the total area occupied by the empty spaces between the sheets.

Assume that an $m \times n$ matrix of characters is used to represent the two sheets, placed side-by-side on the table. Each element of the matrix is either an **X** (corresponding to the gray squares in the figure, representing paper) or a blank (corresponding to the white squares in the figure, indicating space). The first and last columns will always consist of **X**s only (no spaces).

Note that each row of the matrix will contain a **single** stretch of zero or more blanks separating the left sheet from the right sheet.

When the sheets are just touching, the rightmost **X** in some row of the left sheet will be immediately to the left of the leftmost **X** in the same row of the right sheet. Thus, the corresponding matrix representation (also having m rows, but with $n' \leq n$ columns) will have at least one row consisting only of **X**s (no blanks).



3(a)

3(b)

Input format: The first line of the input data will contain m and n , the number of rows and columns. There will be m additional lines, each containing exactly n characters (either **X** or a blank). These lines will begin and end with an **X**, and will have a **single** sequence of zero or more blanks in between.

Output format: Your program should print a single integer, corresponding to the total area occupied by the empty spaces between the sheets, when their rough edges are just touching. Assume that each matrix element (**X** or blank) occupies unit area. For the example shown in the picture, your answer should be 11.

Sample input0: (corresponds to the above figure; all rows have the same number of columns, even though the figure below suggests otherwise)

```
6 13
XXXXX.....XXX
XXX.....XXXXX
XX.....XXX
XXXX...XXXXXX
XXX.....XXXXX
XXXXXX...XXXX
```

Sample output 0: 11

Sample input 1:

```
4 14
XXXX.....XXXXX
XXX.....XXXXXXX
XXXXXX.....XXXX
XX.....XXXXXXX
```

Sample output 1: 4