STEMLoyola Challenges

NO SWEAT ◆ NO GAIN

Welcome-Back Challenge

(Challenge 06)

Begins at **12:00 p.m. EAT** on **Saturday 15th February 2020**

Ends at **11:59 p.m. EAT** on **Sunday 1st March 2020**

The Challenge contains five problems (A, B, C, D, & E) arranged in an increasing order of difficulty. Beginners are expected to solve problems A, B, and C; while experienced students are expected to solve all five problems. Score is determined by both the problems you manage to solve and how sooner you submit a correct solution

Visit contest.stemloyola.org to participate in the challenge and/or view live results

Visit contest.stemloyola.org/register to register, if you do not have an account yet. Registration will be automatically closed when the challenge begins. Hence, ensure you register as soon as possible

Visit challenges.stemloyola.org/article/contest-registration for registration instructions

Visit challenges.stemloyola.org to access various resources (guidelines, tutorials, articles, videos, solutions to previous challenges, etc.) that you may need to complete the challenge

Consult Mr. Albert, Mr. Samuel, or other Computer Department teachers with any questions or concerns you may have

Problem A: Decimal Part

Description

You are given a positive real number with up to ten decimal places. Extract the value of the decimal part.

Input

The input contains the decimal number which is guaranteed to be less than 1,000,000.

Output

Print the value of the decimal part only, when the number is written in four decimal places.

NB: *Kindly note that your solution will be run five times. Each time, it will be tested against a different set of input. The first two test cases are given below to help you check your solution. The remaining tests can be seen from the contest page for this problem or the results page after you submit your solution.*

Test 1

Input	Output
123.456789	0.4568

Test 2

Input	Output
53898.14	0.1400

Problem B: Gotcha!

Description

You are given two numbers whose value is less than 1,000. Evaluate the sum and product of the two numbers and determine which quantity is greater.

Input

The first line of the input contains the first number (`-1,000 < A < 1,000`). The second line of the input contains the second number (`-1,000 < B < 1,000`). **A** and **B** can be integers or decimal numbers with less than five decimal places.

Output

Print "GOTCHA!" to the standard output (i.e. screen) without the quotes, if the sum and product of **A** and **B** are equal. Otherwise, if the sum is greater than the product, print "SUM" without the quotes. If the product is greater than the sum, print "PRODUCT" without the quotes.

NB: *Kindly note that your solution will be run five times. Each time, it will be tested against a different set of input. The first three test cases are given below to help you check your solution. The remaining tests can be seen from the contest page for this problem or the results page after you submit your solution*.

Test 1

Input	Output
2 2	GOTCHA!

Test 2

Input	Output
450.5 0.5	SUM

Test 3

Input	Output
450.5 10	PRODUCT

Problem C: Kiosk Change

Description

During one of his lunch breaks, David visited the school kiosk and spent TZS 1,500 on snacks and drinks. He gave the shopkeeper a 5,000-banknote and received a TZS 3,500 change. His change consisted ot three 1,000-banknotes and one 500-banknote. He received a total of four banknotes. David knew that the shopkeeper could have used fewer banknotes to return the change. The most efficient way to return a TZS 3,500 change is by using one 2,000-banknote, one 1,000-banknote, and one 500-banknote.

You have determined that the school kiosk only uses eight denominations of the Tanzanian shillings (TZS): 50, 100, 200, 500, 1,000, 2,000, 5,000, and 10,000. Can you determine the most efficient way to return a given change? David considered the most efficient way as the one that uses the least amount of banknotes/coins.

Input

The input consists of a single positive integer that denotes a hypothetical change that needs to be returned to a customer. It is guaranteed that the change will be less than TZS 10,000.

Output

Your program should compute the best way to return the change and print how many banknotes/coins will be used.

NB: *Kindly note that your solution will be run five times. Each time, it will be tested against a different set of input. The first two test cases are given below to help you check your solution. The remaining tests can be seen from the contest page for this problem or the results page after you submit your solution*.

Test 1

Input	Output
3500	3

Test 2

Input	Output
8900	6

Problem D: Usernames

Description

Over the holiday, Aisha implemented her first major programming project. The project involved creating a website to help her fellow students practice Basic Mathematics. After letting her friends start using it, she realized that not all of them created unique usernames when they registered. As a result, the website has mixed the details of those whose usernames collide.

Aisha has managed to extract the list of all usernames from the database and has asked for your help to ensure that each username is unique.

Input

The first line in the input consists of an integer (5 <= **N** <= 1000) specifying the number of total usernames in the list. Each of the following **N** lines contains one username extracted from the database.

Output

Your program should first display the total number of ***unique*** usernames in the list. In the following **N** lines, your program should display all username, each on its own line. The usernames should appear in the same order they appeared in the input (so Aisha can import them back into the database). To make each username unique, append 2, 3, 4, etc., for the second-time, third-time, fourth-time, etc., a particular username is used respectively. Refer to ***Test 1*** below for further description of inputs and outputs.

NB: *Kindly note that your solution will be run five times. Each time, it will be tested against a different set of input. The first two test cases are given below to help you check your solution. The remaining tests can be seen from the contest page for this problem or the results page after you submit your solution.*

Test 1

Input	Output
6 Alice Bob Alice Alice Bob Charles	3 Alice Bob Alice2 Alice3 Bob2 Charles

The input for Test 1 contains six usernames (some repeat). The total count of unique usernames is 3. Hence, the output begins by displaying 3. In the second and third lines of the output, Alice and Bob are printed respectively. Since this is the first time these usernames appear, they are written as they are. The fourth and fifth lines of the output contain Alice. Since this is the second and third time that Alice is uses, 2 and 3 are appended to the username as shown. Sixth line of the output contains Bob with 2 appended since it is the second time Bob is used. The seventh line of the output contains Charles, with nothing appended to it since it is the first time it is used.

Test 2

Input	Output
10 Suzie Suzie Mtabe Aisha Jenny Alex Rita Oleta Belinda Danny	9 Suzie Suzie2 Mtabe Aisha Jenny Alex Rita Oleta Belinda Danny

Problem E: Mike's Puzzle

Description

Mike loves puzzles. Since he started learning programming, he has been trying to make programs that solve different kinds of puzzles. Recently, he decided to create his own puzzles. In one of his own creation, he designs a square grid and connects the top left corner to the bottom right corner with a continuous path. To make the puzzle easy to solve, this path does not contain loops or branches. He also guarantees that there is only a single path connecting the two corners.

He then defines the cost of going through the path as the sum of tiles that one passes through (i.e. red/blue tiles) minus the number of times one changes direction. For instance, in ***Figure 1***, to move from the top left corner to the bottom right, one passes through nine (9) tiles and makes three direction changes (at the blue tiles). As a result, Mike defines the cost of this path as six (i.e. 9 tiles minus 3 direction changes). Similarly, in ***Figure 2***, the path has 41 tiles and 9 corners resulting in the cost of 32.

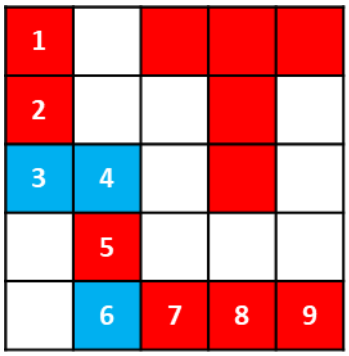

Figure 1

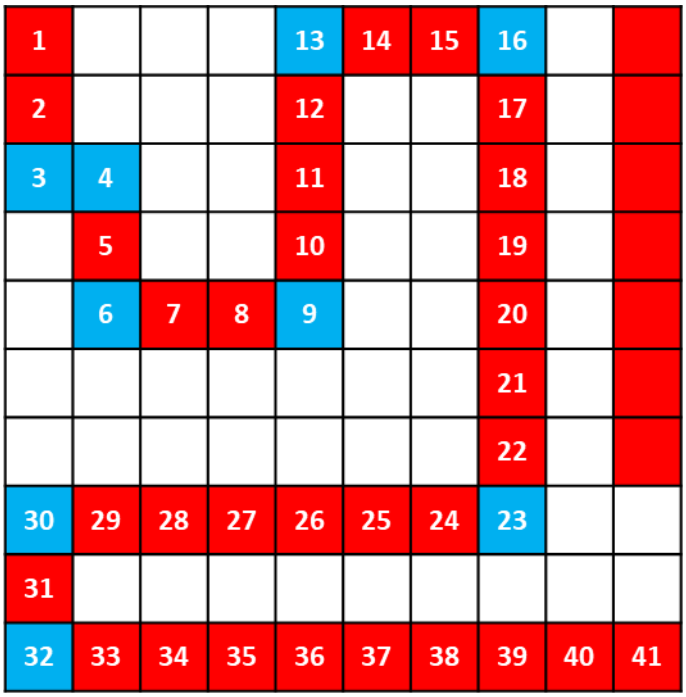

Figure 2

Input

The first line in the input consists of an integer (2 <= **N** <= 50) specifying the size of the grid. The following **N** lines contain the grid details. Each line corresponds to a single row of the grid. Each row contains **N** numbers (each as either a 1 or 0) corresponding to **N** columns of the grid. 1 denotes a path and 0 denotes a wall. The data in ***Test 1*** correspond to the grid in in ***Figure 1***, while the data in ***Test 2*** correspond to the grid in ***Figure 2***.

Output

Your program should compute the effective cost of the available path in a given grid as Mike defines it.

NB: *Kindly note that your solution will be run five times. Each time, it will be tested against a different set of input. The first two test cases are given below to help you check your solution. The remaining tests can be seen from the contest page for this problem or the results page after you submit your solution*.

Test 1

Input	Output
5 1 0 1 1 1 1 0 0 1 0 1 1 0 1 0 0 1 0 0 0 0 1 1 1 1	6

Test 2

Input	Output
10 1 0 0 0 1 1 1 1 0 1 1 0 0 0 1 0 0 1 0 1 1 1 0 0 1 0 0 1 0 1 0 1 0 0 1 0 0 1 0 1 0 1 1 1 1 0 0 1 0 1 0 0 0 0 0 0 0 1 0 1 0 0 0 0 0 0 0 1 0 1 1 1 1 1 1 1 1 1 0 0 1 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1	32