Department of computer Science and Engineering

National Institute of Technology Calicut

Monsoon Semester-2022-23

Evaluation:5 CS 4097D Object-Oriented Systems Lab
Date:19/10/2022

1. You are given a text message for encoding the approach you need to follow is just represent repeated successive characters as the character and a single count

Input Format

The first line contains an integer 'T' which denotes the number of test cases or queries to be run. Then the test cases follow:

The first and only line of each test case will contain a string denoting the message.

Note:The message string will have no digits and consist solely of lowercase alphabetic characters

Output Format

For each test case, print a single line containing the encoded message string.

Output for every test case will be printed in a separate line.

Constraints

1 <= T <= 10
1 <= N <= 100000
Where 'N' is the length of the message string.

Sample INPUT and OUTPUT

3 aabbc abcd abbdcaas	a2b2c1 a1b1c1d1 a1b2d1c1a2s 1

2. Create a class named ***Student*** having 6 instance variables: "name" of type string, "rollno" of type int, "mark1","mark2","mark3" and "total" of type float. Declare 3 methods named "***setStudent()","getTotal()"*** and ***"display()"*** inside the class. ***"setStudent()"***

method is to set the values of the student and ***"getTotal()"*** method is to find and set the value of the total marks. The ***"display()"*** method is to display the details of the student. Create a main class "StudentInfo", read the details of students into an array of student objects and display the rank list based on the ascending order of total marks (If two students have the same total marks, display them in the order of rollno).

Sample INPUT and OUTPUT

```
Enter how many students:
3
Enter details for Student 1
Enter roll no: 53
Enter Name: Sharath
Enter mark 1: 96.25
Enter mark 2: 90.75
Enter mark 3: 88.5

Name: Sharath
Roll No: 53
Mark 1: 96.25
Mark 2: 90.75
Mark 3: 88.5
Total Marks: 275.5

Enter details for Student 2
Enter roll no: 23
Enter Name: Manav
Enter mark 1: 87.75
Enter mark 2: 92.25
Enter mark 3: 85.25

Name: Manav
Roll No: 23
Mark 1: 87.75
Mark 2: 92.25
Mark 3: 85.25
Total Marks: 265.25

Enter details for Student 3
Enter roll no: 15
Enter Name: Ashwin
Enter mark 1: 80
Enter mark 2: 75.75
Enter mark 3: 95.25

Name: Ashwin
```

Roll No: 15
Mark 1: 80.0
Mark 2: 75.75
Mark 3: 95.25
Total Marks: 251.0

Rank:1 - Sharath - 275.5
Rank:2 - Manav - 265.25
Rank:3 – Ashwin - 251.0

3. Create a class named ***'Arithop'***. It has a static instance variable 'nop', representing the number of operations. Also it contains four methods namely ***add()***, ***sub()***, ***mul()*** and ***div()***. The add(), sub() and mul() methods take 2 int type operand arguments and return an int result value. The div() method takes 2 int type operand arguments and returns a double result value. Before returning the result, all methods should increment the value of nop. Create a class '***Cal_Arith***', having only the main function. Inside the main function, continuously read the operand's value and the intended operation as specified in the sample input/output. After the user input, create an object each time and display the result of each arithmetic operation using the functions in the class. After each operation, display the number of operations.

   Sample INPUT and OUTPUT

   Enter input: 5 + 3
   Add : 8
   Value of nop: 1
   Continue(Y/N) : Y

   Enter input: 2 - 2
   Sub : 0
   Value of nop: 2
   Continue(Y/N) : Y

   Enter input: 4 * 5
   Mul : 20
   Value of nop: 3
   Continue(Y/N) : Y

   Enter input: 15 / 4
   Div : 3.75
   Value of nop: 4
   Continue(Y/N) : N

4. There are a number of different kinds of accounts that the bank supports.

- **SavingsAccount-** There is 7% interest paid monthly. There is no minimum balance required. ***Example:)*** *If Balance =50 , added interest balance on the month end is 53.5*
- **CheckingAccount-**
    1. There is 7% interest paid monthly.
    2. This account charges a fee at the end of the month: 10% of the added interest balance.
    3. There is a penalty of 10.00 at the end of the month if the balance (after deducting fee on the added interest balance) falls below a minimum of 100.00.

Create a class **BankAccount**. Create classes **SavingsAccount** and **CheckingAccount** as the subclasses of **BankAccount**. There is a method **getMonthEndBalance** in each of these classes (Identify the type of polymorphism used in the program).

- In BankAccount class, getMonthEndBalance just returns the balance amount
- In SavingsAccount class, getMonthEndBalance returns the added interest balance amount

- In CheckingAccount class, getMonthEndBalance returns the balance amount after applying the interest rate, then applying account charge and then applying the penalty

Input Format: The input contains the accountType and balance amount separated by colon. balance should be>= 20. accountType is either SavingsAccount or CheckingAccount

Output Format: The output prints the result of **getMonthEndBalance** based on the accountType (Final result must be rounded to two decimal places.)

If balance <= 20, then print “overdraft”.

If accountType is not SavingsAccount or CheckingAccount, then print “invalidAccount”

Sample Input 0
CheckingAccount:50

Sample Output 0
38.15

5. Define a class bus in java with the following specifications:

Data Members

- busno - to store Bus Number
- from - to store Place name of origin
- to - to store Place name of destination
- distance - to store the Distance in Kilometres
- type – to store Bus Type such as ‘O’ for ordinary (O, F , L use enum).

Member Functions

- A constructor function to initialize all data members. Use ‘O’ as type if nothing is specified for Type.
- A function CalcFare to calculate Fare as per the following criteria:

Type	Fare
‘O’	10*Distance
‘F’	20*Distance
‘L’	24*Distance

- A function Show to display the content of all the data members on screen

Sample Input/Output:

1-Add details

2-Show details

3-Exit

Sample-0

INPUT

1

120 kannur calicut 500 F (Busno,from,to,distance,type)

1

123 palayam mukkam 1400 O

2

3

OUTPUT

120 kannur calicut 500 F 10000

123 palayam mukkam 1400 O 14000

Sample-1

INPUT

1

601 lucknow agra 300 O

2

1

676 delhi noida 50 F

2

3

OUTPUT

601 lucknow agra 300 O 3000

601 lucknow agra 300 O 3000

676 delhi noida 50 F 1000