

RAJALAKSHMI ENGINEERING COLLEGE
RAJALAKSHMI NAGAR, THANDALAM – 602 105



RAJALAKSHMI
ENGINEERING COLLEGE

An AUTONOMOUS Institution
Affiliated to ANNA UNIVERSITY, Chennai

Laboratory Record Notebook

Name :

Year / Branch / Section :

Register No. :

College Roll No. :

Semester :

Academic Year :



RAJALAKSHMI ENGINEERING COLLEGE

An AUTONOMOUS institution
Affiliated to ANNA UNIVERSITY, Chennai

BONAFIDE CERTIFICATE

NAME :

ACADEMIC YEAR SEMESTER BRANCH

REGISTER NO.

Certified that this is the bonafide record of work done by the above student in the

..... Laboratory during the year 20 - 20

Signature of Faculty - in - Charge

Submitted for the Practical Examination held on

Internal Examiner

External Examiner

INDEX

Name : _____ Branch : _____ Sec : _____ Roll No: _____

[illegible]

Started on	Wednesday, 9 August 2023, 10:30 AM
State	Finished
Completed on	Wednesday, 9 August 2023, 10:59 AM
Time taken	29 mins 11 secs
Marks	3.00/3.00
Grade	15.00 out of 15.00 (100%)
Name	LOSHIKA G CSD

Question 1

Correct

Mark 1.00 out of 1.00

Write a program that returns the last digit of the given number. Last digit is being referred to the least significant digit i.e. the digit in the ones (units) place in the given number.

The last digit should be returned as a positive number.

For example,

if the given number is 197, the last digit is 7

if the given number is -197, the last digit is 7

For example:

Input	Result
197	7
-197	7

Answer: (penalty regime: 0 %)

Ace editor not ready. Perhaps reload page?

Falling back to raw text area.

```
import java.util.*;
public class Sample
{
    public static void main(String args[])
    {
        Scanner scan = new Scanner(System.in);
        int no = scan.nextInt() ;
        no = Math.abs(no);
        System.out.println(no%10);
    }
}
```

	Input	Expected	Got	
✓	197	7	7	✓
✓	-197	7	7	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question **2**

Correct

Mark 1.00 out of 1.00

Write a program to find whether the given input number is Odd.

If the given number is odd, the program should return 2 else It should return 1.

Note: The number passed to the program can either be negative. positive or zero. Zero should NOT be treated as Odd.

For example:

Input	Result
123	2
456	1

Answer: (penalty regime: 0 %)

Ace editor not ready. Perhaps reload page?

Falling back to raw text area.

```
import java.util.*;
public class Sample
{
    public static void main ( String args [] )
    {
        Scanner scan = new Scanner ( System.in ) ;
        int no = scan.nextInt() ;
        no = Math.abs(no);
        System.out.println( (no % 2) +1 );
    }
}
```

	Input	Expected	Got	
✓	123	2	2	✓
✓	456	1	1	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question **3**

Correct

Mark 1.00 out of 1.00

Rohit wants to add the last digits of two given numbers.

For example,

If the given numbers are 267 and 154, the output should be 11.

Below is the explanation:

Last digit of the 267 is 7

Last digit of the 154 is 4

Sum of 7 and 4 = 11

Write a program to help Rohit achieve this for any given two numbers.

Note: Tile sign of the input numbers should be ignored.

i.e.

if the input numbers are 267 and 154, the sum of last two digits should be 11

if the input numbers are 267 and -154, the slim of last two digits should be 11

if the input numbers are -267 and 154, the sum of last two digits should be 11

if the input numbers are -267 and -154, the sum of last two digits should be 11

For example:

Input	Result
267 154	11
267 -154	11
-267 154	11
-267 -154	11

Answer: (penalty regime: 0 %)

Ace editor not ready. Perhaps reload page?

Falling back to raw text area.

```
import java.util.*;
public class Sample
{
    public static void main ( String args [] )
    {
        Scanner scan = new Scanner ( System.in ) ;
        int A = scan.nextInt() ;
        int B = scan.nextInt() ;
        A = Math.abs ( A ) ;
        B = Math.abs ( B ) ;
        System.out.println(( A % 10) + ( B % 10));
    }
}
```

	Input	Expected	Got	
✓	267 154	11	11	✓
✓	267 -154	11	11	✓
✓	-267 154	11	11	✓
✓	-267 -154	11	11	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Started on	Tuesday, 5 March 2024, 8:30 AM
State	Finished
Completed on	Tuesday, 5 March 2024, 9:44 AM
Time taken	1 hour 14 mins
Marks	5.00/5.00
Grade	50.00 out of 50.00 (100%)
Name	LOSHIKA G 2022-CSD-A

Question **1**

Correct

Mark 1.00 out of 1.00

Write a python program that takes a integer between 0 and 15 as input and displays the number of '1' s in its binary form.
(Hint:use python bitwise operator.

Sample Input

3

Sample Output:

2

Explanation:

The binary representation of 3 is 011, hence there are 2 ones in it. so the output is 2.

Answer: (penalty regime: 0 %)

```
1 n=int(input())
2 count1=0
3 b=str(bin(n))
4 for i in b :
5     if i == "1":
6         count1+=1
7 print(count1)
```

	Input	Expected	Got	
✓	3	2	2	✓
✓	5	2	2	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question **2**

Correct

Mark 1.00 out of 1.00

In the 1800s, the battle of Troy was led by Hercules. He was a superstitious person. He believed that his crew can win the battle only if the total count of the weapons in hand is in multiple of 3 and the soldiers are in an even number of count. Given the total number of weapons and the soldier's count, Find whether the battle can be won or not according to Hercules's belief. If the battle can be won print True otherwise print False.

Input format:

Line 1 has the total number of weapons

Line 2 has the total number of Soldiers.

Output Format:

If the battle can be won print True otherwise print False.

Sample Input:

32

43

Sample Output:'

False

Answer: (penalty regime: 0 %)

```
1 a=int(input())
2 b=int(input())
3 print(a%3==0 and b%2==0)
```

	Input	Expected	Got	
✓	32 43	False	False	✓
✓	273 7890	True	True	✓
✓	800 4590	False	False	✓
✓	6789 32996	True	True	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question **3**

Correct

Mark 1.00 out of 1.00

A team from the Rotract club had planned to conduct a rally to create awareness among the Coimbatore people to donate blood. They conducted the rally successfully. Many of the Coimbatore people realized it and came forward to donate their blood to nearby blood banks. The eligibility criteria for donating blood are people should be above or equal to 18 and his/ her weight should be above 40. There was a huge crowd and staff in the blood bank found it difficult to manage the crowd. So they decided to keep a system and ask the people to enter their age and weight in the system. If a person is eligible he/she will be allowed inside.

Write a program and feed it to the system to find whether a person is eligible or not.

Input Format:

Input consists of two integers that correspond to the age and weight of a person respectively.

Output Format:

Display True(IF ELIGIBLE)

Display False (if not eligible)

Sample Input

19

45

Sample Output

True

Answer: (penalty regime: 0 %)

```
1 a=int(input())
2 b=int(input())
3 print((a>17)and(b>40))
```

	Input	Expected	Got	
✓	19 45	True	True	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question 4

Correct

Mark 1.00 out of 1.00

Mr. X's birthday is in next month. This time he is planning to invite N of his friends. He wants to distribute some chocolates to all of his friends after the party. He went to a shop to buy a packet of chocolates. At the chocolate shop, 4 packets are there with different numbers of chocolates. He wants to buy such a packet which contains a number of chocolates, which can be distributed equally among all of his friends. Help Mr. X to buy such a packet.

Input Given:

N-No of friends

P1,P2,P3 AND P4-No of chocolates

OUTPUT:

"True" if he can buy that packet and "False" if he can't buy that packet.

SAMPLE INPUT AND OUTPUT:

5

25

12

10

9

OUTPUT

True False True False

Answer: (penalty regime: 0 %)

```
1 a=int(input())
2 a1=int(input())
3 a2=int(input())
4 a3=int(input())
5 a4=int(input())
6 print(a1%a==0,end=" ")
7 print(a2%a==0,end=" ")
8 print(a3%a==0,end=" ")
9 print(a4%a==0,end=" ")
10
```

	Input	Expected	Got	
✓	5 25 23 20 10	True False True True	True False True True	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question 5

Correct

Mark 1.00 out of 1.00

An online retailer sells two products: widgets and gizmos. Each widget weighs 75 grams. Each gizmo weighs 112 grams. Write a program that reads the number of widgets and the number of gizmos from the user. Then your program should compute and display the total weight of the parts.

Sample Input

10

20

Sample Output

The total weight of all these widgets and gizmos is 2990 grams.

For example:

Input	Result
10 20	The total weight of all these widgets and gizmos is 2990 grams.

Answer: (penalty regime: 0 %)

```
1 a=int(input())
2 b=int(input())
3 ans=(a*75)+(b*112)
4 print("The total weight of all these widgets and gizmos is %d grams."%ans)
```

	Input	Expected	Got	
✓	10 20	The total weight of all these widgets and gizmos is 2990 grams.	The total weight of all these widgets and gizmos is 2990 grams.	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

[◀ Week-2_MCQ](#)

Jump to...

Started on	Wednesday, 6 March 2024, 10:35 AM
State	Finished
Completed on	Wednesday, 6 March 2024, 11:31 AM
Time taken	56 mins 36 secs
Marks	5.00/5.00
Grade	50.00 out of 50.00 (100%)
Name	LOSHIKA G 2022-CSD-A

Question 1

Correct

Mark 1.00 out of 1.00

Write a program that reads an integer from the user. Then your program should display a message indicating whether the integer is even or odd.

Sample Input1:

5

Sample Output1:

5 is odd.

Sample Input2:

10

Sample Output2:

10 is even.

For example:

Input	Result
5	5 is odd.

Answer: (penalty regime: 0 %)

```
1 a=int(input())
2 if(a%2==0):
3     print("%d is even."%a)
4 else:
5     print("%d is odd."%a)
```

	Input	Expected	Got	
✓	5	5 is odd.	5 is odd.	✓
✓	10	10 is even.	10 is even.	✓
✓	20	20 is even.	20 is even.	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question **2**

Correct

Mark 1.00 out of 1.00

Most years have 365 days. However, the time required for the Earth to orbit the Sun is actually slightly more than that. As a result, an extra day, February 29, is included in some years to correct for this difference. Such years are referred to as leap years. The rules for determining whether or not a year is a leap year follow:

- Any year that is divisible by 400 is a leap year.
- Of the remaining years, any year that is divisible by 100 is not a leap year.
- Of the remaining years, any year that is divisible by 4 is a leap year.
- All other years are not leap years.

Write a program that reads a year from the user and displays a message indicating whether or not it is a leap year.

Sample Input 1

1900

Sample Output 1

1900 is not a leap year.

Sample Input 2

2000

Sample Output 2

2000 is a leap year.

Answer: (penalty regime: 0 %)

```
1 a=int(input())
2 if(a%400==0):
3     print(a,"is a leap year.")
4 else:
5     print(a,"is not a leap year.")
```

	Input	Expected	Got	
✓	1900	1900 is not a leap year.	1900 is not a leap year.	✓
✓	2000	2000 is a leap year.	2000 is a leap year.	✓
✓	2100	2100 is not a leap year.	2100 is not a leap year.	✓
✓	2400	2400 is a leap year.	2400 is a leap year.	✓

Passed all tests! ✓

Correct

Question **3**

Correct

Mark 1.00 out of 1.00

Write a Python program that accepts three parameters. The first parameter is an integer. The second is one of the following mathematical operators: +, -, /, or *. The third parameter will also be an integer.

The function should perform a calculation and return the results. For example, if the function is passed 6 and 4, it should return 24.

Sample Input Format:

11

+

14

Sample Output Format:

25

Answer: (penalty regime: 0 %)

```
1 a=int(input())
2 b=input()
3 c=int(input())
4 if b in "+":
5     print(a+c)
6 elif b in "-":
7     print(a-c)
8 elif b in "*":
9     print(a*c)
10 elif b in "/":
11     print(a/c)
12
```

	Input	Expected	Got	
✓	11 + 14	25	25	✓
✓	45 - 50	-5	-5	✓
✓	12 * 100	1200	1200	✓
✓	18 / 2	9.0	9.0	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question 4

Correct

Mark 1.00 out of 1.00

In this exercise you will create a program that reads a letter of the alphabet from the user. If the user enters a, e, i, o or u then your program should display a message indicating that the entered letter is a vowel. If the user enters y then your program should display a message indicating that sometimes y is a vowel, and sometimes y is a consonant. Otherwise your program should display a message indicating that the letter is a consonant.

Sample Input 1

i

Sample Output 1

It's a vowel.

Sample Input 2

y

Sample Output 2

Sometimes it's a vowel... Sometimes it's a consonant.

Sample Input 3

c

Sample Output 3

It's a consonant.

For example:

Input	Result
y	Sometimes it's a vowel... Sometimes it's a consonant.
c	It's a consonant.

Answer: (penalty regime: 0 %)

```
1 a=input()
2 if a in "aeiou" :
3     print("It's a vowel.")
4 elif a in "y":
5     print("Sometimes it's a vowel... Sometimes it's a consonant.")
6 else:
7     print("It's a consonant.")
```

	Input	Expected	Got	
✓	i	It's a vowel.	It's a vowel.	✓
✓	y	Sometimes it's a vowel... Sometimes it's a consonant.	Sometimes it's a vowel... Sometimes it's a consonant.	✓

	Input	Expected	Got	
✓	c	It's a consonant.	It's a consonant.	✓
✓	e	It's a vowel.	It's a vowel.	✓
✓	r	It's a consonant.	It's a consonant.	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question 5

Correct

Mark 1.00 out of 1.00

In the 1800s, the battle of Troy was led by Hercules. He was a superstitious person. He believed that his crew can win the battle only if the total count of the weapons in hand is in multiple of 3 and the soldiers are in an even number of count. Given the total number of weapons and the soldier's count, Find whether the battle can be won or not according to Hercules's belief. If the battle can be won print True otherwise print False.

Input format:

Line 1 has the total number of weapons

Line 2 has the total number of Soldiers.

Output Format:

If the battle can be won print True otherwise print False.

Sample Input:

32

43

Sample Output:'

False

For example:

Input	Result
32 43	False

Answer: (penalty regime: 0 %)

```
1 a=int(input())
2 b=int(input())
3 print((a%3==0)and(b%2==0))
```

	Input	Expected	Got	
✓	32 43	False	False	✓
✓	273 7890	True	True	✓

	Input	Expected	Got	
✓	800 4590	False	False	✓
✓	6789 32996	True	True	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

◀ [Week-03_MCQ](#)

Jump to...

[WEEK-03-Extra](#) ▶

Started on	Friday, 29 March 2024, 10:09 PM
State	Finished
Completed on	Friday, 29 March 2024, 11:08 PM
Time taken	59 mins 17 secs
Marks	5.00/5.00
Grade	50.00 out of 50.00 (100%)
Name	LOSHIKA G 2022-CSD-A

Question 1

Correct

Mark 1.00 out of 1.00

Strong Number:

Strong number is a special number whose sum of factorial of digits is equal to the original number.

For example: 145 is strong number. Since, $1! + 4! + 5! = 145$.

Write a program to find whether the given number is a Strong Number or not.

Input Format:

The Input consists of a single integer n.

Output Format:

Output consists of a single word 'Yes' or 'No'.

Sample Input 1:

145

Sample Output 1:

Yes

Answer: (penalty regime: 0 %)

```
1 def calculate_factorial(n):
2     result=1
3     for i in range(1,n+1):
4         result*=i
5     return result
6 def is_strong_number(number):
7     original_number=number
8     digit_sum=0
9     while number>0:
10        digit=number%10
11        digit_factorial=calculate_factorial(digit)
12        digit_sum+=digit_factorial
13        number//=10
14    return digit_sum==original_number
15 number = int(input())
16 result=is_strong_number(number)
17 if result:
18     print("Yes")
19 else:
20     print("No")
21
```

	Input	Expected	Got	
✓	145	Yes	Yes	✓
✓	40585	Yes	Yes	✓
✓	4321	No	No	✓
✓	2	Yes	Yes	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question **2**

Correct

Mark 1.00 out of 1.00

Write a program to find the sum of the series $1 + 11 + 111 + 1111 + \dots + n$ terms (n will be given as input from the user and sum will be the output)

Sample Test Cases

Test Case 1

Input

4

Output

1234

Explanation:

as input is 4, have to take 4 terms.

$1 + 11 + 111 + 1111$

Test Case 2

Input

6

Output

123456

For example:

Input	Result
3	123

Answer: (penalty regime: 0 %)

```
1 n=int(input())
2 x=1
3 sum=0
4 temp=0
5 for i in range (1,n+1):
6     sum=x+sum
7     temp=temp+sum
8     x=x*10
9 print(temp)
```

	Input	Expected	Got	
✓	1	1	1	✓

	Input	Expected	Got	
✓	3	123	123	✓
✓	4	1234	1234	✓
✓	7	1234567	1234567	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question 3

Correct

Mark 1.00 out of 1.00

Determine the factors of a number (i.e., all positive integer values that evenly divide into a number).

For example:

Input	Result
20	1 2 4 5 10 20

Answer: (penalty regime: 0 %)

```

1 a=int(input())
2 for i in range(1,a+1):
3     if(a%i==0):
4         print(i,end=" ")

```

	Input	Expected	Got	
✓	20	1 2 4 5 10 20	1 2 4 5 10 20	✓
✓	5	1 5	1 5	✓
✓	13	1 13	1 13	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question 4

Correct

Mark 1.00 out of 1.00

Write a program that reads a positive integer, n , from the user and then displays the sum of all of the integers from 1 to n .

Sample Input

10

Sample Output

The sum of the first 10 positive integers is 55.0

For example:

Input	Result
10	The sum of the first 10 positive integers is 55.0

Answer: (penalty regime: 0 %)

```
1 n=int(input())
2 sum=(n*(n+1))/2
3 print("The sum of the first %d positive integers is %.1f"%(n,sum))
```

	Input	Expected	Got	
✓	10	The sum of the first 10 positive integers is 55.0	The sum of the first 10 positive integers is 55.0	✓
✓	20	The sum of the first 20 positive integers is 210.0	The sum of the first 20 positive integers is 210.0	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question 5

Correct

Mark 1.00 out of 1.00

In this exercise you will create a program that computes the average of a collection of values entered by the user. The user will enter 0 as a sentinel value to indicate that no further values will be provided. Your program should display an appropriate error message if the first value entered by the user is 0.

Hint: Because the 0 marks the end of the input it should not be included in the average.

Sample Input

1
2
3
4
5
0

The average is 3.0.

Answer: (penalty regime: 0 %)

```
1 sum=0
2 c=0
3 while True:
4     v=int(input())
5     if(v==0):
6         break
7     sum=sum+v
8     c=c+1
9 print("The average is %.1f"%(sum/c))
```

	Input	Expected	Got	
✓	1 2 3 4 5 0	The average is 3.0.	The average is 3.0.	✓
✓	11 22 33 44 55 0	The average is 33.0.	The average is 33.0.	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

[◀ Week-04_MCQ](#)

Jump to...

[WEEK-04-Extra ▶](#)

Started on	Friday, 29 March 2024, 11:45 PM
State	Finished
Completed on	Friday, 10 May 2024, 12:10 PM
Time taken	41 days 12 hours
Marks	5.00/5.00
Grade	50.00 out of 50.00 (100%)
Name	LOSHIKA G 2022-CSD-A

Question 1

Correct

Mark 1.00 out of 1.00

Given a list and we have to find the index/position of minimum and maximum elements of a list in Python.

```
if list = [10, 1, 2, 20, 3, 20]
```

then it must print

1

20

First line of input is no of elements in a list

Followed by n inputs one by one.

Output line 1 contains index of minimum element

Output line 2 contains index of maximum element

Note: if more than one element is minimum / maximum then first index will be considered.

For example:

Input	Result
3	0
10	1
20	
15	

Answer: (penalty regime: 0 %)

```

1 size=int(input())
2 list=[]
3 for i in range(size):
4     num=int(input())
5     list.append(num)
6 min=max=list[0]
7 min_index=max_index=0
8 for i in range(1,size):
9     if(list[i]<min):
10        min_index=i
11    elif(list[i]>max):
12        max_index=i
13
14 print(min_index)
15 if(max_index-1<0):
16     print("0")
17 else:
18     print(max_index-1)
```

	Input	Expected	Got	
✓	3	0	0	✓
	10	1	1	
	20			
	15			

	Input	Expected	Got	
✓	5	4	4	✓
	12	2	2	
	15			
	85			
	65			
	11			
✓	6	5	5	✓
	6	0	0	
	5			
	4			
	3			
	2			
	1			

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question **2**

Correct

Mark 1.00 out of 1.00

Consider the following program statement:

One needs to first input a set of N number of ALPHABETIC Strings each representing a name of a student in an array studname [N]. Assume each string can be Max. 40 Character Long. Subsequently, one needs to input Marks obtained by those students in another array marks [N]. Assume that studname[I] i.e. ith student in the list of student names has obtained Marks [I] in the Marks List. You need to find out and print the Max Marks obtained by a student and also print the name of the student who has obtained this marks. Considering here both the arrays of size 5. Complete the program by filling up required code in editable section.

Sample Test Cases

Test Case 1

Input

Amit

Bratin

Sandip

Sundar

Patrick

34

48

23

16

45

Output

48

Bratin

Test Case 2

Input

Amit

Bratin

Sandip

Sundar

Patrick

49

48

34

23

45

Output

49

Amit

For example:

Input	Result
Amit	90
Bratin	Bratin
Sandip	
Sundar	
Patrick	
89	
90	
45	
67	
82	

Answer: (penalty regime: 0 %)

```
1 studname = []
2 marks = []
3
4 for i in range(5):
5     name = input()
6     studname.append(name)
7 for i in range(5):
8     mark = int(input())
9     marks.append(mark)
10 max_marks = max(marks)
11 max_index = marks.index(max_marks)
12 max_student = studname[max_index]
13 print(max_marks)
14 print(max_student)
```

	Input	Expected	Got	
✓	Amit Bratin Sandip Sundar Patrick 89 90 45 67 82	90 Bratin	90 Bratin	✓
✓	Amit Bratin Sandip Sundar Patrick 34 48 23 16 45	48 Bratin	48 Bratin	✓

	Input	Expected	Got	
✓	Amit Bratin Sandip Sundar Patrick 49 48 34 23 45	49 Amit	49 Amit	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question **3**

Correct

Mark 1.00 out of 1.00

Write a program that reads integers from the user and stores them in a list. Use 0 as a sentinel value to mark the end of the input. Once all of the values have been read your program should display them (except for the 0) in reverse order, with one value appearing on each line.

Sample Input

```
33
11
22
55
44
0
```

Sample Output

```
55
44
33
22
11
```

For example:

Input	Result
33	55
11	44
22	33
55	22
44	11
0	

Answer: (penalty regime: 0 %)

```
1 integer_list = []
2 while True:
3     num = int(input())
4     if num == 0:
5         break
6     integer_list.append(num)
7 integer_list.sort(reverse=True)
8 for num in integer_list:
9     print(num)
```

	Input	Expected	Got	
✓	33 11 22 55 44 0	55 44 33 22 11	55 44 33 22 11	✓
✓	50 40 20 10 30 0	50 40 30 20 10	50 40 30 20 10	✓
✓	1 2 3 4 5 6 7 8 9 0	9 8 7 6 5 4 3 2 1	9 8 7 6 5 4 3 2 1	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question **4**

Correct

Mark 1.00 out of 1.00

Create a program that reads integers from the user until a -99 is entered. Once all of the integers have been read your program should display all of the negative numbers, followed by all of the zeros, followed by all of the positive numbers. Within each group, the numbers should be displayed in the same order that they were entered by the user. For example, if the user enters the values 3, -4, 1, 0, -1, 0, and -2 then your program should output the values -4, -1, -2, 0, 0, 3, and 1. Your program should display each value on its own line. (-99 is not included in the final display)

Sample Input

0
5
10
-15
-20
-99

Sample Output

-15
-20
0
5
10

For example:

Input	Result
0	-15
5	-20
10	0
-15	5
-20	10
-99	

Answer: (penalty regime: 0 %)

```
1 def main():
2     negatives = []
3     zeros = []
4     positives = []
5     while True:
6         num = int(input())
7         if num == -99:
8             break
9         elif num < 0:
10            negatives.append(num)
11        elif num == 0:
12            zeros.append(num)
13        else:
14            positives.append(num)
15    for num in negatives:
16        print(num)
17    for num in zeros:
```

```
18 |         print(num)
19 |     for num in positives:
20 |         print(num)
21 |
22 | if __name__ == "__main__":
```

	Input	Expected	Got	
✓	0	-15	-15	✓
	5	-20	-20	
	10	0	0	
	-15	5	5	
	-20	10	10	
	-99			
✓	10	-40	-40	✓
	20	-50	-50	
	30	0	0	
	-40	10	10	
	-50	20	20	
	0	30	30	
	-99			

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question **5**

Incorrect

Mark 1.00 out of 1.00

An array is monotonic if it is either **monotone increasing** or **monotone decreasing**.

An array A is monotone increasing if for all $i \leq j$, $A[i] \leq A[j]$. An array A is monotone decreasing if for all $i \leq j$, $A[i] \geq A[j]$.

Write a program if n array is monotonic or not. Print "True" if is monotonic or "False" if it is not. Array can be monotone increasing or decreasing.

Input Format:

First line n-get number of elements

Next n Lines is the array of elements

Output Format:

True ,if array is monotone increasing or decreasing.

otherwise False is printed

Sample Input1

4
5
6
7
8

Sample Output1

True

Sample Input2

4
6
5
4
3

Sample Output2

True

Sample Input 3

4
6
7
8
7

Sample Output3

False

For example:

Input	Result
4 6 5 4 3	True

Answer: (penalty regime: 0 %)

```

1 def is_monotonic(arr):
2     increasing = decreasing = True
3     for i in range(1, len(arr)):
4         if arr[i] < arr[i - 1]:
5             increasing = False
6         if arr[i] > arr[i - 1]:
7             decreasing = False
8     return increasing or decreasing
9 n = int(input())
10 arr = [int(input()) for _ in range(n)]
11 result = is_monotonic(arr)
12 print(result)
13

```

	Input	Expected	Got	
✓	4 6 5 4 3	True	True	✓
✗	4 3 5 7 9	False	True	✗

Your code must pass all tests to earn any marks. Try again.

Show differences

Incorrect

Marks for this submission: 0.00/1.00.

◀ [Week-05_MCQ](#)

Jump to...

[WEEK-05-Extra](#) ▶

Started on	Monday, 22 April 2024, 1:30 PM
State	Finished
Completed on	Monday, 22 April 2024, 2:32 PM
Time taken	1 hour 1 min
Marks	5.00/5.00
Grade	50.00 out of 50.00 (100%)
Name	LOSHIKA G 2022-CSD-A

Question 1

Correct

Mark 1.00 out of 1.00

Verify the given number is cyclic or not.

Input Format

Num1

Num2

Constraints $1 \leq \text{range} \leq 9999999999$ **Sample Input 1**

12345

45123

Sample Output 1

Yes

Sample Input 2

12345

54123

Sample Output 2

No

Answer: (penalty regime: 0 %)

```
1 def is_cyclic(num1,num2):
2     num1_str = str(num1)
3     num2_str = str(num2)
4
5     if len(num1_str) != len(num2_str):
6         return "No"
7
8     if num1_str in num2_str + num2_str:
9         return "Yes"
10    else:
11        return "No"
12
13 num1 = int(input())
14 num2 = int(input())
15 print(is_cyclic(num1,num2))
16
```

	Input	Expected	Got	
✓	12345 45123	Yes	Yes	✓

	Input	Expected	Got	
✓	12345 54123	No	No	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question **2**

Correct

Mark 1.00 out of 1.00

Write a code to reverse the case of a character input

Input Format:

Single character Input

Output Format:

Reversed character

Example Input:

R

Output:

r

Example Input:

a

Output:

A

For example:

Input	Result
R	r
a	A

Answer: (penalty regime: 0 %)

```
1 a=input()
2 print(a.swapcase())
```

	Input	Expected	Got	
✓	R	r	r	✓
✓	a	A	A	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question **3**

Correct

Mark 1.00 out of 1.00

Consider the below words as key words and check the given input is key word or not.

keywords: {break, case, continue, default, defer, else, for, func, goto, if, map, range, return, struct, type, var}

Input format:

Take string as an input from stdin.

Output format:

Print the word is key word or not.

Example Input:

break

Output:

break is a keyword

Example Input:

IF

Output:

IF is not a keyword

For example:

Input	Result
break	break is a keyword
IF	IF is not a keyword

Answer: (penalty regime: 0 %)

```
1 str1=str(input())
2 key=['break' , 'case' , 'continue','default','defer','else','for','goto','if','map','range
3 if str1 in key:
4     print("%s is a keyword"%(str1))
5 else:
6     print("%s is not a keyword"%(str1))
```

	Input	Expected	Got	
✓	break	break is a keyword	break is a keyword	✓
✓	IF	IF is not a keyword	IF is not a keyword	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question 4

Correct

Mark 1.00 out of 1.00

Find if a String2 is substring of String1. If it is, return the index of the first occurrence. else return -1.

Sample Input 1

thistest123string

123

Sample Output 1

8

Answer: (penalty regime: 0 %)

```
1 def find_substring_index(string1,string2):
2     index = string1.find(string2)
3     return index
4 string1 = str(input())
5 string2 = str(input())
6
7 index = find_substring_index(string1,string2)
8 if index != -1:
9     print(index)
10 else:
11     print("Not found ")
```

	Input	Expected	Got	
✓	thistest123string 123	8	8	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question 5

Correct

Mark 1.00 out of 1.00

Write a program to get 3 strings as input.

In the 1st string, replace the vowels with "

In the 2nd string, replace the consonants with *

In the third string, convert the lowercase letters to upper case.

Input Format:

Take 3 Strings from stdin

Output Format:

- In the 1st string, replace the vowels with "
- In the 2nd string, replace the consonants with *
- In the third string, convert the lowercase letters to upper case.

Example Input:

Hello

Hi

GoodMorning

Output:

H"ll"

*i

GOODMORNING

Answer: (penalty regime: 0 %)

```

1 def replace_vowels_with_quotes(string):
2     vowels = "aeiouAEIOU"
3     return ''.join(['' if char in vowels else char for char in string])
4
5 def replace_consonants_with_asterisks(string):
6     consonants = "bcdfghjklmnpqrstvwxyzBCDFGHJKLMNPQRSTVWXYZ"
7     return ''.join(['*' if char in consonants else char for char in string])
8
9 def convert_to_uppercase(string):
10    return string.upper()
11
12 def main():
13     # Input three strings
14     string1 = input()
15     string2 = input()
16     string3 = input()
17
18     # Perform operations
19     string1_modified = replace_vowels_with_quotes(string1)
20     string2_modified = replace_consonants_with_asterisks(string2)
21     string3_modified = convert_to_uppercase(string3)
22

```

	Input	Expected	Got	
✓	Hello Hi GoodMorning	H"ll" *i GOODMORNING	H"ll" *i GOODMORNING	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

◀ Week-06_MCQ

Jump to...

WEEK-06-Extra ▶

Started on	Monday, 22 April 2024, 2:07 PM
State	Finished
Completed on	Monday, 22 April 2024, 3:55 PM
Time taken	1 hour 48 mins
Marks	5.00/5.00
Grade	50.00 out of 50.00 (100%)
Name	LOSHIKA G 2022-CSD-A

Question 1

Correct

Mark 1.00 out of 1.00

A string with parentheses is well bracketed if all parentheses are matched: every opening bracket has a matching closing bracket and vice versa.

Write a Python function `wellbracketed(s)` that takes a string `s` containing parentheses and returns `True` if `s` is well bracketed and `False` otherwise.

Hint: Keep track of the nesting depth of brackets. Initially the depth is 0. The depth increases with each opening bracket and decreases with each closing bracket. What are the constraints on the value of the nesting depth for the string to be wellbracketed?

Here are some examples to show how your function should work.

```
>>> wellbracketed("22")
```

```
False
```

```
>>> wellbracketed("(a+b)(a-b)")
```

```
True
```

```
>>> wellbracketed("(a(b+c)-d)((e+f)")
```

```
False
```

Answer: (penalty regime: 0 %)

[Reset answer](#)

```
1 def wellbracketed(s):
2     count = 0
3     open_string = "("
4     close_string = ")"
5
6     for char in s :
7         if(char in open_string):
8             count += 1
9         elif(char in close_string):
10            count -= 1
11        else:
12            continue
13    if count == 0:
14        return True
15    else:
16        return False
```

	Test	Expected	Got	
✓	<code>print(wellbracketed("22"))</code>	False	False	✓
✓	<code>print(wellbracketed("(a+b)(a-b)"))</code>	True	True	✓
✓	<code>print(wellbracketed("(a(b+c)-d)((e+f)"))</code>	False	False	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question **2**

Correct

Mark 1.00 out of 1.00

Write a program that reads values from the user until a blank line is entered. Display the total of all of the values entered by the user (or 0 if the first value entered is a blank line). Complete this task using recursion. Your program may not use any loops.

Hint: The body of your recursive function will need to read one value from the user, and then determine whether or not to make a recursive call. Your function does not need to take any arguments, but it will need to return a numeric result.

Sample Input

5
10
15
20
25

Sample Output

75

Answer: (penalty regime: 0 %)

Reset answer

```
1 def readAndTotal():
2     value = input("")
3     if not value:
4         return 0
5     else:
6         try:
7             num = float(value)
8             return num + readAndTotal()
9         except ValueError:
10            print("Invalid input.Please enter a number.")
11            return get_total()
12
13 total = readAndTotal()
14 print("%.0f"%total)
15
16
```

	Input	Expected	Got	
✓	5 10 15 20 25	75	75	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question **3**

Correct

Mark 1.00 out of 1.00

A prime number is an integer greater than one that is only divisible by one and itself. Write a function that determines whether or not its parameter is prime, returning True if it is, and False otherwise.

Answer: (penalty regime: 0 %)

Reset answer

```
1 def isPrime(n):
2     if n<=1:
3         return False
4     elif n<=3:
5         return True
6     elif n%2==0 or n%3==0:
7         return False
8     i=5
9     while i*i<=n:
10        if n%i==0 or n%(i+2)==0:
11            return False
12        i+=6
13    return True
```

	Test	Expected	Got	
✓	print(isPrime(1))	False	False	✓
✓	print(isPrime(2))	True	True	✓
✓	print(isPrime(3))	True	True	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question 4

Correct

Mark 1.00 out of 1.00

Write a Python function `sumofsquares(m)` that takes an integer `m` returns `True` if `m` is a sum of squares and `False` otherwise. (If `m` is not positive, your function should return `False`.)

Here are some examples to show how your function should work.

```
>>> sumofsquares(41)
```

```
True
```

```
>>> sumofsquares(30)
```

```
False
```

```
>>> sumofsquares(17)
```

```
True
```

Answer: (penalty regime: 0 %)

[Reset answer](#)

```
1 from math import sqrt
2 def issquare(n):
3     k = int(sqrt(n))
4     return k * k == n
5 def sumofsquares(m):
6     if m <= 0:
7         return False
8     for i in range(1, int(sqrt(m)) + 1):
9         if issquare(m - i*i):
10            return True
11     return False
12
```

	Test	Expected	Got	
✓	<code>print(sumofsquares(41))</code>	True	True	✓
✓	<code>print(sumofsquares(30))</code>	False	False	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question 5

Correct

Mark 1.00 out of 1.00

The notion of a palindrome was introduced previously. In this exercise you will write a recursive function that determines whether or not a string is a palindrome. The empty string is a palindrome, as is any string containing only one character. Any longer string is a palindrome if its first and last characters match, and if the string formed by removing the first and last characters is also a palindrome.

Write a program that reads a string from the user and uses your recursive function to determine whether or not it is a palindrome. Then your program should display an appropriate message for the user.

Sample Input

malayalam

Sample Output

That was a palindrome!

Sample Input

madan

Sample Output

That is not a palindrome.

Answer: (penalty regime: 0 %)

Reset answer

```
1 def isPalindrome(s):
2     if len(s) <= 1:
3         return True
4     else:
5         return s[0] == s[-1] and isPalindrome(s[1:-1])
6
7 text = input("")
8 if isPalindrome(text):
9     print("That was a palindrome!")
10 else:
11     print("That is not a palindrome.")
12
13
```

	Input	Expected	Got	
✓	malayalam	That was a palindrome!	That was a palindrome!	✓
✓	madan	That is not a palindrome.	That is not a palindrome.	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Jump to...

WEEK-07-Extra ►

Started on	Thursday, 30 May 2024, 10:09 PM
State	Finished
Completed on	Thursday, 30 May 2024, 10:12 PM
Time taken	3 mins 24 secs
Marks	5.00/5.00
Grade	50.00 out of 50.00 (100%)
Name	LOSHIKA G 2022-CSD-A

Question 1

Correct

Mark 1.00 out of 1.00

Write a program to read a string and a character and find the whether the character is available in the string or not. Print True if the character is present in the string, False otherwise.

Sample Input

Rakalakshmi

a

Sample Output

True

Sample Input

Rakalakshmi

b

Sample Output

False

Answer: (penalty regime: 0 %)

```
1 s=input()
2 a=input()
3 if a in s:
4     print(True)
5 else:
6     print(False)
```

	Input	Expected	Got	
✓	Rajalakshmi a	True	True	✓
✓	Rajalakshmi b	False	False	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question **2**

Correct

Mark 1.00 out of 1.00

Write a python program to read a string and a character, print the number of occurrence of the character in the string and the location of the first occurrence.

Note: To convert an input string to tuple use tuple(variablename).

Sample Input

Apple

p

Sample Output

2

1

Answer: (penalty regime: 0 %)

```
1 def character_info(s, c):
2     s_tuple = tuple(s)
3     count = s_tuple.count(c)
4
5     if count > 0:
6         first_occurrence = s_tuple.index(c)
7     else:
8         first_occurrence = -1
9
10    print(count)
11    print(first_occurrence)
12
13    import sys
14    input_data = sys.stdin.read().strip().split()
15    input_string = input_data[0]
16    input_char = input_data[1]
17
18    character_info(input_string, input_char)
```

	Input	Expected	Got	
✓	Apple p	2 1	2 1	✓
✓	Rajalakshmi a	3 1	3 1	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question **3**

Correct

Mark 1.00 out of 1.00

Create different types of tuples as per below-mentioned values and print the same.

```
()  
(4, 5, 8)  
(1, 'ECE', 'MCT', 'R&A', 3.4)  
( 'Python', [8, 4, 6], (1, 2, 3))
```

Answer: (penalty regime: 0 %)

```
1 t1 = ()  
2 t2 = (4, 5, 6)  
3 t3 = (1, 'ECE', 'MCT', 'R&A', 3.4)  
4 t4 = ( 'Python', [8, 4, 6], (1, 2, 3))  
5  
6 print(t1)  
7 print(t2)  
8 print(t3)  
9 print(t4)
```

	Expected	Got	
✓	<pre>() (4, 5, 6) (1, 'ECE', 'MCT', 'R&A', 3.4) ('Python', [8, 4, 6], (1, 2, 3))</pre>	<pre>() (4, 5, 6) (1, 'ECE', 'MCT', 'R&A', 3.4) ('Python', [8, 4, 6], (1, 2, 3))</pre>	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question **4**

Correct

Mark 1.00 out of 1.00

Rahul went to a supermarket to buy some product, he has purchased the products and about to pay the bill, where the items he purchased is been stored in a nested tuples in the following order ((item_name,item_cost,no_of_item)), consider raju has purchased 5 items, calculate the total cost for the items he purchased.

sample input:

bread

45

5

milk

40

2

cheese

60

2

butter

90

2

jam

60

2

sample output: 725

Answer: (penalty regime: 0 %)

```
1 def calculate_total(items):
2     total_cost = 0
3     for item in items:
4         total_cost += item[1] * item[2]
5     print(total_cost)
6
7 items = [(input(), int(input()), int(input())) for _ in range(5)]
8 calculate_total(items)
```

	Input	Expected	Got	
✓	bread 45 5 milk 40 2 cheese 60 2 butter 90 2 jam 60 2	725	725	✓
✓	noodles 55 5 egg 10 10 ketchup 80 2 cooldrinks 100 2 fruit 160 2	1055	1055	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question **5**

Correct

Mark 1.00 out of 1.00

Write a python program to count the no. of Occurrence of an item in the tuple and print the list of items and no. of Occurrence more than one time in sorted order.

Input formate:

10 numbers in 10 lines

Sample Input:

50

70

40

60

70

50

80

60

20

60

Sample Output:

50:2

60:3

70:2

Answer: (penalty regime: 0 %)

```
1 # Step 1: Read 10 numbers from input
2 numbers = []
3 for _ in range(10):
4     num = int(input())
5     numbers.append(num)
6
7 # Step 2: Store the numbers in a tuple
8 numbers_tuple = tuple(numbers)
9
10 # Step 3: Count the occurrences of each number in the tuple
11 occurrences = {}
12 for num in numbers_tuple:
13     if num in occurrences:
14         occurrences[num] += 1
15     else:
16         occurrences[num] = 1
17
18 # Step 4: Filter out the numbers that occur more than once
19 filtered_occurrences = {num: count for num, count in occurrences.items() if count > 1}
20
21 # Step 5: Sort the numbers based on their values
22 sorted_numbers = sorted(filtered_occurrences.items())
```

	Input	Expected	Got	
✓	50	50:2	50:2	✓
	70	60:3	60:3	
	40	70:2	70:2	
	60			
	70			
	50			
	80			
	60			
	20			
	60			
✓	40	10:2	10:2	✓
	50	30:3	30:3	
	30	40:2	40:2	
	60			
	30			
	20			
	40			
	10			
	30			
	10			

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

◀ Week-08_MCQ

Jump to...

Week-09_MCQ ▶

Started on	Thursday, 30 May 2024, 9:59 PM
State	Finished
Completed on	Friday, 31 May 2024, 5:06 AM
Time taken	7 hours 7 mins
Marks	5.00/5.00
Grade	50.00 out of 50.00 (100%)
Name	LOSHIKA G 2022-CSD-A

Question 1

Correct

Mark 1.00 out of 1.00

Two strings, a and b , are called anagrams if they contain all the same characters in the same frequencies. For example, the anagrams of CAT are CAT, ACT, TAC, TCA, ATC, and CTA.

Complete the function in the editor. If a and b are case-insensitive anagrams, print "Anagrams"; otherwise, print "Not Anagrams" instead.

Input Format

The first line contains a [string](#) denoting a .

The second line contains a [string](#) denoting b .

Constraints

- $1 \leq \text{length}(a), \text{length}(b) \leq 50$
- Strings a and b consist of English alphabetic characters.
- The comparison should NOT be case sensitive.

Output Format

Print "Anagrams" if a and b are case-insensitive anagrams of each other; otherwise, print "Not Anagrams" instead.

Sample Input 0

anagram

margana

Sample Output 0

Anagrams

Explanation 0

Character	Frequency: anagram	Frequency: margana
A or a	3	3
G or g	1	1
N or n	1	1
M or m	1	1
R or r	1	1

The two strings contain all the same letters in the same frequencies, so we print "Anagrams".

Answer: (penalty regime: 0 %)

```
1 def is_anagram(a, b):
2     # Convert both strings to lowercase
3     a = a.lower()
4     b = b.lower()
5
6     # Sort the characters in both strings
7     sorted_a = sorted(a)
8     sorted_b = sorted(b)
9
10    # Check if the sorted strings are equal
11    if sorted_a == sorted_b:
12        return "Anagrams"
13    else:
14        return "Not Anagrams"
15
16    # Read input strings
17    a = input().strip()
18    b = input().strip()
19
20    # Check if strings are anagrams
21    result = is_anagram(a, b)
22    print(result)
```

	Input	Expected	Got	
✓	madam maDaM	Anagrams	Anagrams	✓
✓	DAD DAD	Anagrams	Anagrams	✓
✓	MAN MAM	Not Anagrams	Not Anagrams	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question **2**

Correct

Mark 1.00 out of 1.00

Given a sorted linked list, delete all duplicates such that each element appear only *once*.

Example 1:**Input:**

1 1 2

Output:

1 2

Example 2:**Input:**

1 1 2 3 3

Output:

1 2 3

Answer: (penalty regime: 0 %)

```
1 arr = input().split()
2 list1=set(map(int,arr))
3 list=sorted(list1)
4 for num in list:
5     print(num,end=" ")
```

	Test	Input	Expected	Got	
✓	1	1 1 2	1 2	1 2	✓
✓	2	1 1 2 3 3	1 2 3	1 2 3	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question **3**

Correct

Mark 1.00 out of 1.00

Take a complete sentence as an input and remove duplicate word in it and print (sorted order), then count all the words which have a length greater than 3 and print.

Input

we are good are we good

Output

are good we

Count = 1

For example:

Input	Result
welcome to rec rec cse ece	cse ece rec to welcome Count = 1

Answer: (penalty regime: 0 %)

```
1 s = input().strip()
2 words = s.split()
3 unique_words = sorted(set(words))
4 print(' '.join(unique_words))
5 count = sum(1 for word in unique_words if len(word) > 3)
6 print("Count = %s"%count)
```

	Input	Expected	Got	
✓	we are good are we good	are good we Count = 1	are good we Count = 1	✓
✓	welcome to rec rec cse ece	cse ece rec to welcome Count = 1	cse ece rec to welcome Count = 1	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question **4**

Correct

Mark 1.00 out of 1.00

You are given an array of N integers, A1, A2, . . . , AN and an integer K. Return the of count of distinct numbers in all windows of size K.

Input :

1 2 1 3 4 3

3

Output :

2

3

3

2

Explanation

All windows of size K are

[1, 2, 1]

[2, 1, 3]

[1, 3, 4]

[3, 4, 3]

Answer: (penalty regime: 0 %)

```
1 def countDistinct(arr, k):
2     result = []
3     window = {}
4     distinct_count = 0
5
6     # Count distinct elements in the first window
7     for i in range(k):
8         if arr[i] not in window:
9             window[arr[i]] = 1
10            distinct_count += 1
11        else:
12            window[arr[i]] += 1
13
14    result.append(distinct_count)
15
16    # Slide the window and update distinct count for each window
17    for i in range(k, len(arr)):
18        if window[arr[i - k]] == 1:
19            distinct_count -= 1
20            window[arr[i - k]] -= 1
21
22        if arr[i] not in window or window[arr[i]] == 0:
```

	Input	Expected	Got	
✓	1 2 1 3 4 3 3	2 3 3 2	2 3 3 2	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question 5

Correct

Mark 1.00 out of 1.00

Mr.Harish is maintaining a phone directory which stores phone numbers. He will update the directory with phone numbers every week. While entering the input the number should not be stored inside if the phone number already exists. Finally he want his phone number to be printed in ascending order

Input: n – A1 array size and m – A2 arraysze

Array A1 containing phone numbers already existing and Array A2 containing numbers to be inserted

Ouput : Phone numbers printed in ascending order

Sample Test Case

Input

5

6

9840403212 9890909012 98123455 90123456 99123456

90909090 99999999 9840403212 12345678 12347890 99123456

Output

12345678 12347890 90123456 90909090 98123455 99123456 99999999 9840403212 9890909012

Answer: (penalty regime: 0 %)

```

1 n = int(input())
2 m = int(input())
3 A1 = list(map(int, input().split()))
4 A2 = list(map(int, input().split()))
5 set_A1 = set(A1)
6 set_A2 = set(A2)
7 combined_set = set_A1.union(set_A2)
8 sorted_numbers = sorted(combined_set)
9 print(" ".join(map(str, sorted_numbers)))
10

```

	Input	Expected	Got	
✓	3 3 9876543211 1122334455 6677889911 6677889911 9876543211 4455667788	1122334455 4455667788 6677889911 9876543211	1122334455 4455667788 6677889911 9876543211	✓
✓	5 6 9840403212 9890909012 98123455 90123456 99123456 90909090 99999999 9840403212 12345678 12347890 99123456	12345678 12347890 90123456 90909090 98123455 99123456 99999999 9840403212 9890909012	12345678 12347890 90123456 90909090 98123455 99123456 99999999 9840403212 9890909012	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

◀ Week-09_MCQ

Jump to...

WEEK-09-Extra ▶

Started on	Thursday, 30 May 2024, 9:43 PM
State	Finished
Completed on	Thursday, 30 May 2024, 9:58 PM
Time taken	15 mins 40 secs
Marks	7.00/7.00
Grade	50.00 out of 50.00 (100%)
Name	LOSHIKA G 2022-CSD-A

Question 1

Correct

Mark 1.00 out of 1.00

In the game of Scrabble™, each letter has points associated with it. The total score of a word is the sum of the scores of its letters. More common letters are worth fewer points while less common letters are worth more points. The points associated with each letter are shown below:

Points Letters

1 A, E, I, L, N, O, R, S, T and U

2 D and G

3 B, C, M and P

4 F, H, V, W and Y

5 K

8 J and X

10 Q and Z

Write a program that computes and displays the Scrabble™ score for a word. Create a dictionary that maps from letters to point values. Then use the dictionary to compute the score.

A Scrabble™ board includes some squares that multiply the value of a letter or the value of an entire word. We will ignore these squares in this exercise.

Sample Input

REC

Sample Output

REC is worth 5 points.

Answer: (penalty regime: 0 %)

```

1 def scrabble_score(word):
2     letter_scores = {
3         'AEILNORSTU': 1,
4         'DG': 2,
5         'BCMP': 3,
6         'FHVWY': 4,
7         'K': 5,
8         'JX': 8,
9         'QZ': 10
10    }
11
12    score = 0
13    for char in word.upper():
14        for letters, value in letter_scores.items():
15            if char in letters:
16                score += value
17                break
18
19    return score
20
21 word = input()
22 score = scrabble_score(word)

```

	Input	Expected	Got	
✓	REC	REC is worth 5 points.	REC is worth 5 points.	✓
✓	RAJALAKSHMI	RAJALAKSHMI is worth 27 points.	RAJALAKSHMI is worth 27 points.	✓

Passed all tests! ✓

Correct

Question **2**

Correct

Mark 1.00 out of 1.00

Multiply All the Items in a Dictionary

Input: Any input in Dictionary format (Ex: d={'A':10,'B':10,'C':239})

Output: multiplication of dictionary values (23900)

Answer: (penalty regime: 0 %)

```
1 d = {'A':10,'B':10,'C':239}
2 result = 1
3 for value in d.values():
4     result *= value
5 print(result)
```

	Input	Expected	Got	
✓	d={'A':10, 'B':10, 'C':239}	23900	23900	✓

Passed all tests! ✓

Correct

Question **3**

Correct

Mark 1.00 out of 1.00

A teacher wants to evaluate her class results for the subject she handles. She want to do the following analysis:

1. Display Class average
2. Display Maximum mark Roll no
3. Display Minimum mark Roll no

Kindly help her out. Use dictionary for storing the student details.

Input Format:

In line 1 no of students will be given

Followed by n lines containing student rollno and marks

Output Format:

Line 1 Class average

Line 2 Maximum mark Roll no

Line 3 Minimum mark Roll no

Sample Input:

```
4
01 87
02 99
03 45
04 77
```

Output:

```
77
02
03
```

Answer: (penalty regime: 0 %)

```
1 n = int(input())
2 marks = {}
3
4 for _ in range(n):
5     roll, mark = input().split()
6     marks[int(mark)] = int(roll)
7
8 class_average = sum(marks.keys()) / n
9
10 max_mark = max(marks)
11 max_roll = marks[max_mark]
12 min_mark = min(marks)
13 min_roll = marks[min_mark]
14
15 print(int(class_average))
16 print("%d"%max_roll)
17 print("%d"%min_roll)
```


	Input	Expected	Got	
✓	4 01 87 02 99 03 45 04 77	77 02 03	77 02 03	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question **4**

Correct

Mark 1.00 out of 1.00

To Check if a Given Key Exists in a Dictionary or Not

Input: Any dictionary format input (Ex: d={'A':1,'B':2,'C':3})

Enter Key to check: A

Output:

Key is present and value of the key is: (location)

Present # True Statement

Not Present # False Statement

Answer: (penalty regime: 0 %)

```
1 def c(d, k):
2     if k in d:
3         print("Present")
4     else:
5         print("Not Present")
6
7 d = {'A': 1, 'B': 2, 'C': 3}
8 k = input()
9
10 c(d, k)
```

	Input	Expected	Got	
✓	A	Present	Present	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question 5

Correct

Mark 1.00 out of 1.00

Create a program that determines and displays the number of unique characters in a string entered by the user. For example, Hello, World! has 10 unique characters while zzz has only one unique character. Use a dictionary or set to solve this problem.

For example:

Input	Result
Hello, World!	10

Answer: (penalty regime: 0 %)

```
1 def unique_characters_count(s):
2     char_count = {}
3     for char in s:
4         if char in char_count:
5             char_count[char] += 1
6         else:
7             char_count[char] = 1
8     return len(char_count)
9
10 user_input = input()
11 print(unique_characters_count(user_input))
```

	Input	Expected	Got	
✓	Hello, World!	10	10	✓
✓	zzz	1	1	✓
✓	RECCSE	4	4	✓
✓	AAABBBCCC	3	3	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question 6

Correct

Mark 1.00 out of 1.00

A sentence is a list of words that are separated by a single space with no leading or trailing spaces. Each word consists of lowercase and uppercase English letters.

A sentence can be shuffled by appending the 1-indexed word position to each word then rearranging the words in the sentence.

For example, the sentence "This is a sentence" can be shuffled as "sentence4 a3 is2 This1" or "is2 sentence4 This1 a3".

Given a shuffled sentence *s* containing no more than 9 words, reconstruct and return the original sentence.

Example 1:

Input:

is2 sentence4 This1 a3

Output:

This is a sentence

Explanation: Sort the words in *s* to their original positions "This1 is2 a3 sentence4", then remove the numbers.

Example 2:

Input:

Myself2 Me1 I4 and3

Output:

Me Myself and I

Explanation: Sort the words in *s* to their original positions "Me1 Myself2 and3 I4", then remove the numbers.

Constraints:

2 <= *s*.length <= 200

s consists of lowercase and uppercase English letters, spaces, and digits from 1 to 9.

The number of words in *s* is between 1 and 9.

The words in *s* are separated by a single space.

s contains no leading or trailing spaces.

Answer: (penalty regime: 0 %)

```
1 def reconstruct_sentence(s):
2     words = s.split()
3     word_positions = {}
4     for word in words:
5         position = int(word[-1])
6         word_without_position = word[:-1]
7         word_positions[position] = word_without_position
8     sorted_words = [word_positions[i] for i in range(1, len(words) + 1)]
9     reconstructed_sentence = ' '.join(sorted_words)
10    return reconstructed_sentence
11
12 shuffled_sentence = input()
13 print(reconstruct_sentence(shuffled_sentence))
```

	Input	Expected	Got	
✓	is2 sentence4 This1 a3	This is a sentence	This is a sentence	✓
✓	Myself2 Me1 Vijay4 and3	Me Myself and Vijay	Me Myself and Vijay	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.



Question 7

Correct

Mark 1.00 out of 1.00

Two words are anagrams if they contain all of the same letters, but in a different order. For example, "evil" and "live" are anagrams because each contains one "e", one "i", one "l", and one "v". Create a program that reads two strings from the user, determines whether or not they are anagrams, and reports the result.

Sample Input 1

evil

live

Sample Output 1

Those strings are anagrams.

Sample Input 2

meet

met

Sample Output 2

Those strings are not anagrams.

Answer: (penalty regime: 0 %)

```
1 def are_anagrams(s1, s2):
2     return sorted(s1) == sorted(s2)
3
4 s1 = input().strip()
5 s2 = input().strip()
6
7 if are_anagrams(s1, s2):
8     print("Those strings are anagrams.")
9 else:
10    print("Those strings are not anagrams.")
```

	Input	Expected	Got	
✓	evil live	Those strings are anagrams.	Those strings are anagrams.	✓
✓	meet met	Those strings are not anagrams.	Those strings are not anagrams.	✓
✓	rec cer	Those strings are anagrams.	Those strings are anagrams.	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

[◀ Week-10_MCQ](#)

Jump to...

[WEEK-10-Extra ▶](#)