Dashboard / My courses / CD19411-PPD-2022 / WEEK_01-Introduction to python-Variables-Datatypes-Input/Output-Formatting / Week-1_CODING

Started on	Wednesday, 28 February 2024, 10:22 AM
State	Finished
Completed on	Wednesday, 28 February 2024, 11:23 AM
Time taken	1 hour
Marks	5.00/5.00
Grade	**50.00** out of 50.00 (**100**%)
Name	KANIMOZHI S 2022-CSD-A

Question **1**

Correct

Mark 1.00 out of 1.00

Justin is a carpenter who works on an hourly basis. He works in a company where he is paid Rs 50 for an hour on weekdays and Rs 80 for an hour on weekends. He works 10 hrs more on weekdays than weekends. If the salary paid for him is given, write a program to find the number of hours he has worked on weekdays and weekends.

Hint:

If the final result(hrs) are in -ve convert that to +ve using abs() function

The `abs()` function returns the absolute value of the given number.

```
number = -20
absolute_number = abs(number)
print(absolute_number)
# Output: 20
```

Sample Input:

450

Sample Output:

weekdays 10.38

weekend 0.38

For example:

Input	Result
450	weekdays 10.38 weekend 0.38

Answer: (penalty regime: 0 %)

```
hrs=int(input())
weken=abs((hrs-500))/130
week=weken+10
print("weekdays %.2f"%(week))
print("weekend %.2f"%(weken))
```

	Input	Expected	Got	
✓	450	weekdays 10.38 weekend 0.38	weekdays 10.38 weekend 0.38	✓
✓	500	weekdays 10.00 weekend 0.00	weekdays 10.00 weekend 0.00	✓
✓	10000	weekdays 83.08 weekend 73.08	weekdays 83.08 weekend 73.08	✓

	Input	Expected	Got	
✓	6789	weekdays 58.38 weekend 48.38	weekdays 58.38 weekend 48.38	✓

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 convert strings to an integer and float and display its type.

Sample Input:

10

10.9

Sample Output:

10,<class 'int'>

10.9,<class 'float'>

Answer: (penalty regime: 0 %)

```
n=input()
n3=input()
n1=int(n)
n2=float(n3)

print(n1,type(n1),sep=",")
print("%.1f"%(n2),type(n2),sep=",",end="")
```

	Input	Expected	Got	
✓	10 10.9	10,<class 'int'> 10.9,<class 'float'>	10,<class 'int'> 10.9,<class 'float'>	✓
✓	12 12.5	12,<class 'int'> 12.5,<class 'float'>	12,<class 'int'> 12.5,<class 'float'>	✓
✓	89 7.56	89,<class 'int'> 7.6,<class 'float'>	89,<class 'int'> 7.6,<class 'float'>	✓
✓	55000 56.2	55000,<class 'int'> 56.2,<class 'float'>	55000,<class 'int'> 56.2,<class 'float'>	✓
✓	2541 2541.679	2541,<class 'int'> 2541.7,<class 'float'>	2541,<class 'int'> 2541.7,<class 'float'>	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question **3**

Correct

Mark 1.00 out of 1.00

In many jurisdictions, a small deposit is added to drink containers to encourage people to recycle them. In one particular jurisdiction, drink containers holding one liter or less have a $0.10 deposit and drink containers holding more than one liter have a $0.25 deposit. Write a program that reads the number of containers of each size(less and more) from the user. Your program should continue by computing and displaying the refund that will be received for returning those containers. Format the output so that it includes a dollar sign and always displays exactly two decimal places.

Sample Input

10

20

Sample Output

Your total refund will be $6.00.

For example:

Input	Result
20 20	Your total refund will be $7.00.

Answer: (penalty regime: 0 %)

```
n1=int(input())
n2=int(input())
n3=float(n1*(0.10)+n2*(0.25))
print("Your total refund will be $%.2f."%(n3))

```

	Input	Expected	Got	
✓	20 20	Your total refund will be $7.00.	Your total refund will be $7.00.	✓
✓	11 22	Your total refund will be $6.60.	Your total refund will be $6.60.	✓
✓	123 200	Your total refund will be $62.30.	Your total refund will be $62.30.	✓
✓	76 38	Your total refund will be $17.10.	Your total refund will be $17.10.	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question **4**

Correct

Mark 1.00 out of 1.00

Alfred buys an old scooter for Rs. X and spends Rs. Y on its repairs. If he sells the scooter for Rs. Z (Z>X+Y). Write a program to help Alfred to find his gain percent. Get all the above-mentioned values through the keyboard and find the gain percent.

Input Format:

The first line contains the Rs X

The second line contains Rs Y

The third line contains Rs Z

Sample Input:

10000

250

15000

Sample Output:

46.34 is the gain percent.

For example:

Input	Result
10000 250 15000	46.34 is the gain percent.

Answer: (penalty regime: 0 %)

```
x=int(input())
y=int(input())
z=int(input())
s=x+y
g=((z-s)/s)*100
print("%.2f is the gain percent."%g)
```

	Input	Expected	Got	
✓	10000 250 15000	46.34 is the gain percent.	46.34 is the gain percent.	✓
✓	45500 500 60000	30.43 is the gain percent.	30.43 is the gain percent.	✓
✓	5000 0 7000	40.00 is the gain percent.	40.00 is the gain percent.	✓

	Input	Expected	Got	
✓	12500 5000 18000	`2.86 is the gain percent.`	`2.86 is the gain percent.`	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question **5**

Correct

Mark 1.00 out of 1.00

Write a simple python program to find the square root of a given floating point number. The output should be displayed with 3 decimal places.

Sample Input:

8.00

Sample Output:

2.828

For example:

Input	Result
8.00	2.828

Answer: (penalty regime: 0 %)

```
import math

n=float(input())
print("%.3f"%(math.sqrt(n)))

```

	Input	Expected	Got	
✓	8.00	2.828	2.828	✓
✓	14.00	3.742	3.742	✓
✓	4.00	2.000	2.000	✓
✓	487	22.068	22.068	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

◄ Week-1_MCQ

Jump to...

Week1 extra ►

Dashboard / My courses / CD19411-PPD-2022 / WEEK 02-Operators in Python / WEEK-02 CODING

Started on	Tuesday, 5 March 2024, 8:09 AM
State	Finished
Completed on	Tuesday, 5 March 2024, 8:44 AM
Time taken	34 mins 56 secs
Marks	5.00/5.00
Grade	**50.00** out of 50.00 (**100**%)
Name	KANIMOZHI S 2022-CSD-A

Question **1**

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 %)

```
w=int(input())
s=int(input())
if (w%3==0 and s%2==0):
    print("True")
else:
    print("False")
```

	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 **2**

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 %)

```
a=int(input())
w=int(input())
if a>=18 and w>40:
    print("True")
else:
    print("False")
```

	Input	Expected	Got	
✓	19 45	True	True	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question **3**

Correct

Mark 1.00 out of 1.00

Mr.Ram has been given a problem kindly help him to solve it. The input of the program is either 0 or 1. IF 0 is the input he should display "C" if 1 is the input it should display "D".There is a constraint that Mr. Ram should use either logical operators or arithmetic operators to solve the problem, not anything else.

Hint:

Use ASCII values of C and D.

Input Format:

An integer x, 0<=x<=1. .

Output Format:

output a single character "C" or "D"depending on the value of x.

```
Input 1:
0
Output 1:
C
```

```
Input 2:
1

Output 1:
D
```

Answer: (penalty regime: 0 %)

```
n=int(input())
if n==0:
    print("C")
else:
    print("D")

```

	Input	Expected	Got	
✓	0	C	C	✓
✓	1	D	D	✓

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 %)

```
n=int(input())

for i in range(4):
    p=int(input())
    if p%n==0:
        print("True",end=" ")
    else:
        print("False",end=" ")
```

	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

Pretend that you have just opened a new savings account that earns 4 percent interest per year. The interest that you earn is paid at the end of the year, and is added to the balance of the savings account. Write a program that begins by reading the amount of money deposited into the account from the user. Then your program should compute and display the amount in the savings account after 1, 2, and 3 years. Display each amount so that it is rounded to 2 decimal places.

Sample Input:

10000

Sample Output:

Balance as of end of Year 1: $10400.00.

Balance as of end of Year 2: $10816.00.

Balance as of end of Year 3: $11248.64.

Answer: (penalty regime: 0 %)

```
n=int(input())
n1=n+((n*4)/100)
n2=n1+((n1*4)/100)
n3=n2+((n2*4)/100)
print("Balance as of end of Year 1: $%.2f."%((n1)))
print("Balance as of end of Year 2: $%.2f."%((n2)))
print("Balance as of end of Year 3: $%.2f."%((n3)))

```

	Input	Expected	Got	
✓	10000	Balance as of end of Year 1: $10400.00. Balance as of end of Year 2: $10816.00. Balance as of end of Year 3: $11248.64.	Balance as of end of Year 1: $10400.00. Balance as of end of Year 2: $10816.00. Balance as of end of Year 3: $11248.64.	✓
✓	20000	Balance as of end of Year 1: $20800.00. Balance as of end of Year 2: $21632.00. Balance as of end of Year 3: $22497.28.	Balance as of end of Year 1: $20800.00. Balance as of end of Year 2: $21632.00. Balance as of end of Year 3: $22497.28.	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

◄ Week-2_MCQ

Jump to...

WEEK-02 Extra ►

Dashboard / My courses / CD19411-PPD-2022 / WEEK_03-Selection Structures in Python / WEEK-03_CODING

Started on	Wednesday, 6 March 2024, 10:18 AM
State	Finished
Completed on	Wednesday, 6 March 2024, 11:07 AM
Time taken	48 mins 41 secs
Marks	5.00/5.00
Grade	**50.00** out of 50.00 (**100**%)
Name	KANIMOZHI S 2022-CSD-A

Question **1**

Correct

Mark 1.00 out of 1.00

IN / OUT

Ms. Sita, the faculty handling programming lab for you is very strict. Your seniors have told you that she will not allow you to enter the week's lab if you have not completed atleast half the number of problems given last week. Many of you didn't understand this statement and so they requested the good programmers from your batch to write a program to find whether a student will be allowed into a week's lab given the number of problems given last week and the number of problems solved by the student in that week.

Input Format:

Input consists of 2 integers.

The first integer corresponds to the number of problems given and the second integer corresponds to the number of problems solved.

Output Format:

Output consists of the string “IN” or “OUT”.

Sample Input and Output:

Input

8

3

Output

OUT

For example:

Input	Result
8 3	OUT

Answer: (penalty regime: 0 %)

```
p=int(input())
s=int(input())
if s>=p/2:
    print("IN")
else:
    print("OUT")
```

	Input	Expected	Got	
✓	8 3	OUT	OUT	✓
✓	8 5	IN	IN	✓
✓	20 9	OUT	OUT	✓
✓	50 31	IN	IN	✓

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

For example:

Input	Result
32 43	False

Answer: (penalty regime: 0 %)

```
w=int(input())
s=int(input())
if w%3==0 and s%2==0:
    print("True")
else:
    print("False")
```

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

	Input	Expected	Got	
✓	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

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 %)

```
n1=int(input())
ch=input()
n2=int(input())

if ch=="+":
        print(n1+n2)
elif ch=="-":
        print(n1-n2)
elif ch=="/":
        print(n1/n2)
elif ch=="*":
        print(n1*n2)

```

	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

Write a program to calculate and print the Electricity bill where the unit consumed by the user is given from test case. It prints the total amount the customer has to pay. The charge are as follows:

Unit	Charge / Unit
Upto 199	@1.20
200 and above but less than 400	@1.50
400 and above but less than 600	@1.80
600 and above	@2.00

If bill exceeds Rs.400 then a surcharge of 15% will be charged and the minimum bill should be of Rs.100/-

Sample Test Cases

Test Case 1

Input

50

Output

100.00

Test Case 2

Input

300

Output

517.50

For example:

Input	Result
100.00	120.00

Answer: (penalty regime: 0 %)

```
n=float(input())
if(n<=199.00):
   m=1.20*n
   if(m<100):
     print("100.00")
   else:
     print("%.2f"%(m))
elif n>=200.00 and n<400.00:
    print("%.2f"%(1.50*n))
elif n>=400.00 and n<600.00:
    print("%.2f"%(1.80*n+(15/100)*1.80*n))
elif n>=600.00:
    print("%.2f"%(2.00*n+(15/100)*2.00*n))
```

	Input	Expected	Got	
✓	50	100.00	100.00	✓

	Input	Expected	Got	
✓	100.00	120.00	120.00	✓
✓	500	1035.00	1035.00	✓
✓	700	1610.00	1610.00	✓

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 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 %)

```
n=int(input())
if n%2==0:
    print("%d is even."%(n))
else:
    print("%d is odd."%(n))
```

	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.

◄ Week-03_MCQ

Jump to...

WEEK-03-Extra ►

Dashboard / My courses / CD19411-PPD-2022 / WEEK_04-Iteration Control Structures-LOOPING / WEEK-04_CODING

Started on	Friday, 22 March 2024, 10:19 AM
State	Finished
Completed on	Wednesday, 27 March 2024, 12:23 PM
Time taken	5 days 2 hours
Marks	5.00/5.00
Grade	**50.00** out of 50.00 (**100**%)
Name	KANIMOZHI S 2022-CSD-A

Question **1**

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 %)

```
sum=0
count=0
n=1
while(n!=0):
    n=int(input())
    if(n==0):
        break
    else:
        sum=sum+n
        count=count+1
avg=sum/ count
print("The average is %.1f."%(avg))
```

	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.

Question **2**

Correct

Mark 1.00 out of 1.00

You are choreographing a circus show with various animals. For one act, you are given two kangaroos on a number line ready to jump in the positive direction.

•The first kangaroo starts at position x1 and moves at a speed v1 meters per jump.

•The second kangaroo starts at position x2 and moves at a speed of v2 meters per jump and x2 > x1

•You have to figure out to get both kangaroos at the same position at the same time as part of the show before k jumps. If it is possible, return YES, otherwise return NO.

Input Format:

x1-position of kangaroo1

v1-Speed of kangaroo1

x2-position of kangaroo2

v2-Speed of kangaroo2

k-jumps

Output Format:

Both kangaroos are at the same position within k jumps, YES, otherwise NO.

For example:

Input	Result
0 3 4 2 6	YES

Answer: (penalty regime: 0 %)

```
x1=int(input())
v1=int(input())
x2=int(input())
v2=int(input())
k=int(input())
value=v1*v2
if(value==k):
  print("YES")
else:
    print("NO")
```

	Input	Expected	Got	
✓	0 3 4 2 6	YES	YES	✓
✓	0 3 2 4 8	NO	NO	✓

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 to find the count of ALL digits in a given number N. The number will be passed to the program as an input of type int.

Assumption: The input number will be a positive integer number>= 1 and<= 25000.

For e.g.

If the given number is 292, the function should return 3 because there are 3 digits in this number

If the given number is 1015, the function should return 4 because there are 4 digits in this number

For example:

InputResult

292 3

1015 4

For example:

Input	Result
293	3

Answer: (penalty regime: 0 %)

```
n=int(input())
count=0
while(n!=0):
    n=n//10
    count=count+1
print(count)

```

	Input	Expected	Got	
✓	293	3	3	✓
✓	6788	4	4	✓
✓	52321	5	5	✓

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 %)

```
n=int(input())
sum=0.0
for i in range(1,n+1):
    sum=sum+i
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

Write a program to return the nth number in the fibonacci series.

The value of N will be passed to the program as input.

NOTE: Fibonacci series looks like –

0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, . . . and so on.

i.e. Fibonacci series starts with 0 and 1, and continues generating the next number as the sum of the previous two numbers.

- first Fibonacci number is 0,
- second Fibonacci number is 1,
- third Fibonacci number is 1,
- fourth Fibonacci number is 2,
- fifth Fibonacci number is 3,
- sixth Fibonacci number is 5,
- seventh Fibonacci number is 8, and so on.

For example:

Input:

7

Output

8

For example:

Input	Result
8	13

Answer: (penalty regime: 0 %)

```
n=int(input())
count=2
n1=0
n2=1
n3=1
while(True):
    n3=n1+n2
    n1=n2
    n2=n3
    count=count+1
    if(count==n):
        print(n3)
        break
```

	Input	Expected	Got	
✓	4	2	2	✓

	Input	Expected	Got	
✓	8	13	13	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

◄ Week-04_MCQ

Jump to...

WEEK-04-Extra ►

Dashboard / My courses / CD19411-PPD-2022 / WEEK 05-Lists / WEEK-05 CODING

Started on	Wednesday, 20 March 2024, 10:15 AM
State	Finished
Completed on	Wednesday, 20 March 2024, 11:33 AM
Time taken	1 hour 18 mins
Marks	5.00/5.00
Grade	**50.00** out of 50.00 (**100**%)
Name	KANIMOZHI S 2022-CSD-A

Question **1**

Correct

Mark 1.00 out of 1.00

A teacher in a school entered marks in an array. But mistakenly the teacher repeated the marks twice in between the array. Help the teacher to find how many elements are duplicated in an array

Input:

n – number of elements and the elements to be stored in an array.

Output:

d- number of duplicate elements

Sample Test Case

Input

8

21 35 56 67 67 89 89 90

Output

2

Explanation

The numbers 67 and 89 are repeated , so count is 2

Answer: (penalty regime: 0 %)

```
def count_duplicates(marks):
    unique_marks = set()
    duplicate_count = 0
    for mark in marks:
        if mark in unique_marks:
            duplicate_count += 1
        else:
            unique_marks.add(mark)
    return duplicate_count

# Input
n = int(input())

marks = list(map(int, input().split()))

# Output
print(count_duplicates(marks))

```

	Input	Expected	Got	
✓	8 21 35 56 67 67 89 89 90	2	2	✓
✓	12 56 56 78 78 90 90 95 97 97 99 99 89	5	5	✓
✓	4 67 67 89 90	1	1	✓

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 that takes two lists and returns True if they have at least one common member.

First line of input contains List 1

Second line of input contains List 2

Output is True if there is atleast one common element, false if no common elements

For example:

Input	Result
10 20 30 40 50 12 25 85 40 21	True

Answer: (penalty regime: 0 %)

```
lst1=[]
lst2=[]
k=0
lst1 = list(map(int, input().split()))
lst2 = list(map(int, input().split()))
n=len(lst1)
for i in range(n):
    if lst1[i]==lst2[i]:
        k=1
        print("True")
        break
if(k==0):
   print("False")

```

	Input	Expected	Got	
✓	10 20 30 40 50 12 25 85 40 21	True	True	✓
✓	1 2 3 4 5 7 8 9 10 11	False	False	✓
✓	10 20 30 20 20 30	True	True	✓

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. Your program should continue reading values until the user enters 0. Then it should display all of the values entered by the user (except for the 0) in ascending order, with one value appearing on each line. Use either the sort method or the sorted function to sort the list.

Sample Input

```
20
30
40
50
10
0
```

```
Sample Output
```

```
10
20
30
40
50
```

For example:

Input	Result
20 30 40 50 10 0	10 20 30 40 50

Answer: (penalty regime: 0 %)

```
l=[]
while(True):
   n=int(input())
   if n==0:
     break
   l.append(n)
l.sort()
for i in range(len(l)):
    print(l[i])

```

	Input	Expected	Got	
✓	20 30 40 50 10 0	10 20 30 40 50	10 20 30 40 50	✓
✓	22 33 44 11 55 0	11 22 33 44 55	11 22 33 44 55	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question **4**

Correct

Mark 1.00 out of 1.00

Program to print all the distinct elements in an array. Distinct elements are nothing but the unique (non-duplicate) elements present in the given array.

Input Format:

First line take an Integer input from stdin which is array length n.

Second line take n Integers which is inputs of array.

Output Format:

Print the Distinct Elements in Array in single line which is space Separated

Example Input:

5

1 2 2 3 4

Output:

1 2 3 4

Example Input:

6

1 1 2 2 3 3

Output:

1 2 3

For example:

Input	Result
5 1 2 2 3 4	1 2 3 4

Answer: (penalty regime: 0 %)

```
n = int(input())
arr=[]
arr1=[]
for i in range(n):
   arr.append(int(input()))
arr1=list(set(arr))

for i in arr1:
   print(i,end=" ")

```

	Input	Expected	Got	
✓	5 1 2 2 3 4	1 2 3 4	1 2 3 4	✓
✓	6 1 1 2 2 3 3	1 2 3	1 2 3	✓
✓	5 11 22 11 22 11	11 22	11 22	✓
✓	10 1 2 3 4 5 1 2 3 4 5	1 2 3 4 5	1 2 3 4 5	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question **5**

Correct

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 <= j, **A[i] <= A[j]**. An array A is monotone decreasing if for all i <= j, **A[i] >= 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 %)

```
def is_monotonic(arr):
    increasing = decreasing = True

    # Check if the array is monotonic increasing
    for i in range(len(arr) - 1):
        if arr[i] > arr[i + 1]:
            increasing = False

    # Check if the array is monotonic decreasing
    for i in range(len(arr) - 1):
        if arr[i] < arr[i + 1]:
            decreasing = False

    # If either increasing or decreasing is True, array is monotonic
    return increasing or decreasing

# Input
n = int(input())
arr = [int(input()) for _ in range(n)]

# Output
print(is_monotonic(arr))
```

	Input	Expected	Got	
✓	4 6 5 4 3	True	True	✓
✓	4 3 5 7 9	True	True	✓
✓	4 1 6 9 2	False	False	✓
✓	4 9 6 4 2	True	True	✓
✓	3 2 1 4	False	False	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

◄ Week-05_MCQ

Jump to...

WEEK-05-Extra ►

Dashboard / My courses / CD19411-PPD-2022 / WEEK 06-Strings / WEEK-06 CODING

Started on	Wednesday, 3 April 2024, 11:47 AM
State	Finished
Completed on	Saturday, 13 April 2024, 5:22 AM
Time taken	9 days 17 hours
Marks	5.00/5.00
Grade	**50.00** out of 50.00 (**100**%)
Name	KANIMOZHI S 2022-CSD-A

Question **1**

Correct

Mark 1.00 out of 1.00

Given a string s consisting of some words separated by some number of spaces, return the length of the last word in the string.

A word is a maximal substring consisting of non-space characters only.

For example:

Input	Result
`Hello World`	5
`fly me   to   the moon`	4

Answer: (penalty regime: 0 %)

```
n=input()
k=n.split(" ")
s=k[len(k)-1]

print(len(s))
```

	Input	Expected	Got	
✓	Hello World	5	5	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question **2**

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 %)

```
keywords=["break","case","continue","default","defer","else","for","func","goto","i
n=input()
k=0
for i in range(len(keywords)):
    if keywords[i]==n:
        print(n,"is a keyword")
        k=1
        break
if(k==0):
    print(n,"is not a keyword")
```

	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 **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 %)

```
# Define the set of keywords
keywords = {'break', 'case', 'continue', 'default', 'defer', 'else', 'for', 'func',

# Take input from user
input_word = input()  # Convert the input to lowercase for case-insensitive compari

# Check if the input word is a keyword
if input_word in keywords:
    print(input_word, "is a keyword")
else:
    print(input_word, "is not a keyword")


```

	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

Write a Python program to get one string and reverses a string. The input string is given as an array of characters `char[]` .

You may assume all the characters consist of printable ascii characters.

Example 1:

```
Input:
hello
Output:
olleh
```

Example 2:

```
Input:
Hannah
Output:
hannaH
```

Answer: (penalty regime: 0 %)

```python
def reverse_string(s):
    left = 0
    right = len(s) - 1
    while left < right:
        s[left], s[right] = s[right], s[left]
        left += 1
        right -= 1

# Example usage:
input_str = list(input())
reverse_string(input_str)
print(''.join(input_str))

```

	Input	Expected	Got	
✓	hello	olleh	olleh	✓
✓	Hannah	hannaH	hannaH	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question **5**

Correct

Mark 1.00 out of 1.00

String should contain only the words are not palindrome.

Sample Input 1

Malayalam is my mother tongue

Sample Output 1

is my mother tongue

Answer: (penalty regime: 0 %)

```
def is_palindrome(word):
    return word == word[::-1]

def filter_non_palindromic_words(sentence):
    non_palindromic_words = []
    words = sentence.split()
    for word in words:
        if not is_palindrome(word.lower()):  # Convert word to lowercase before che
            non_palindromic_words.append(word)
    return ' '.join(non_palindromic_words)

# Sample Input
input_string = input()

# Sample Output
output_string = filter_non_palindromic_words(input_string)
print( output_string)


```

	Input	Expected	Got	
✓	Malayalam is my mother tongue	is my mother tongue	is my mother tongue	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

◄ Week-06_MCQ

Jump to...

WEEK-06-Extra ►

Dashboard / My courses / CD19411-PPD-2022 / WEEK_07-Functions / WEEK-07_CODING

Started on	Thursday, 9 May 2024, 11:58 AM
State	Finished
Completed on	Thursday, 9 May 2024, 12:35 PM
Time taken	37 mins 34 secs
Marks	5.00/5.00
Grade	**50.00** out of 50.00 (**100**%)
Name	KANIMOZHI S 2022-CSD-A

Question **1**

Correct

Mark 1.00 out of 1.00

Euclid was a Greek mathematician who lived approximately 2,300 years ago. His algorithm for computing the greatest common divisor of two positive integers, a and b, is both efficient and recursive. It is outlined below:

```
If b is 0 then
    eturn a
Else
    Set c equal to the remainder when a is divided by b
    Return the greatest common divisor of b and c
```

Write a program that implements Euclid's algorithm and uses it to determine the greatest common divisor of two integers entered by the user. Test your program with some very large integers. The result will be computed quickly, even for huge numbers consisting of hundreds of digits, because Euclid's algorithm is extremely efficient.

Answer: (penalty regime: 0 %)

```python
def gcd(a,b):
    if b==0:
        return a
    else:
        c=a%b
        return gcd(b,c)
a=int(input())
b=int(input())
x=gcd(a,b)
print(x)
```

	Input	Expected	Got	
✓	8 12	4	4	✓
✓	720 1000	40	40	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question **2**

Correct

Mark 1.00 out of 1.00

In this exercise you will write a function that determines whether or not a password is good. We will define a good password to be a one that is at least 8 characters long and contains at least one uppercase letter, at least one lowercase letter, and at least one number. Your function should return True if the password passed to it as its only parameter is good. Otherwise it should return False. Include a main program that reads a password from the user and reports whether or not it is good. Ensure that your main program only runs when your solution has not been imported into another file.

Sample Input 1

chennai

Sample Output 1

That isn't a good password.

Sample Input 2

Chennai18

Sample Output 2

That's a good password.

Answer: (penalty regime: 0 %)

Reset answer

```python
def checkPassword(input1):
    l=1
    k=1
    u=1
    if len(input1)>=8:
        for i in input1:
            if i.isupper():
                 k=1
            if i.islower() and k==1:
                l=2
            if i.isnumeric() and l==2:
                u=3
                break
    if u==3:
        print("That's a good password.")
    else:
        print("That isn't a good password.")


```

	Test	Expected	Got	
✓	checkPassword('chennai')	That isn't a good password.	That isn't a good password.	✓
✓	checkPassword('Chennai18')	That's a good password.	That's a good password.	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question **3**

Correct

Mark 1.00 out of 1.00

Write a function that takes three numbers as parameters, and returns the median value of those parameters as its result.

Answer: (penalty regime: 0 %)

Reset answer

```
def median(a, b, c):
    l=[a,b,c]
    l.sort()
    return(l[1])
```

	Test	Expected	Got	
✓	`print(median(10, 20, 30))`	20	20	✓
✓	`print(median(60, 50, 40))`	50	50	✓
✓	`print(median(70, 90, 80))`	80	80	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question **4**

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

```
def wellbracketed(s):
    l=[]
    count=0
    for i in s:
        if i=='(':
           count=count+1
        if i==')':
            count=count-1
    if count==0:
        return True
    else:
        return False
```

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

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question **5**

Correct

Mark 1.00 out of 1.00

Given an integer n, return an list of length n + 1 such that for each i (0 <= i <= n), ans[i] is the number of 1's in the binary representation of i.

Example:

```
Input: n = 2
Output: [0,1,1]
Explanation:
0 --> 0
1 --> 1
2 --> 10
```

Example2:

```
Input: n = 5
Output: [0,1,1,2,1,2]
Explanation:
0 --> 0
1 --> 1
2 --> 10
3 --> 11
4 --> 100
5 --> 101
```

Note: Complete the given function alone

For example:

Test	Result
print(CountingBits(5))	[0, 1, 1, 2, 1, 2]

Answer: (penalty regime: 0 %)

Reset answer

```python
def CountingBits(n):
    l=[]
    for i in range(n+1):
        s=bin(i)
        k=s.count("1")
        l.append(k)
    return l
```

	Test	Expected	Got	
✓	print(CountingBits(2))	[0, 1, 1]	[0, 1, 1]	✓
✓	print(CountingBits(5))	[0, 1, 1, 2, 1, 2]	[0, 1, 1, 2, 1, 2]	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

◄ Week-07_MCQ

Jump to...

WEEK-07-Extra ►

Dashboard / My courses / CD19411-PPD-2022 / WEEK_08-Tuple / WEEK-08_CODING

Started on	Friday, 3 May 2024, 12:01 PM
State	Finished
Completed on	Friday, 3 May 2024, 12:42 PM
Time taken	40 mins 36 secs
Marks	5.00/5.00
Grade	**50.00** out of 50.00 (**100**%)
Name	KANIMOZHI S 2022-CSD-A

Question **1**

Correct

Mark 1.00 out of 1.00

Write a python program to find the the total and average of the students mark. print the total and average of each student as tuple.

Input: first line no.of student, next n * 4 line student marks(four lines for each tuple)

3

20

30

35

45

30

54

60

45

50

60

70

75

Output:

Total : (130,189,255)

Average : (32.50,47.25,63.75)

For example:

Input	Result
3 20 30 35 45 30 54 60 45 50 60 70 75	Total : (130, 189, 255) Average : (32.5, 47.25, 63.75)

Answer: (penalty regime: 0 %)

```
n=int(input())
sum=0
t=()
a=()
avg=0
for i in range(n):
    sum=0
    avg=0
    for j in range(4):
        m=int(input())
        sum=sum+m

    t=t+(sum,)
    avg=sum/4
    a=a+(avg,)
print("Total :",t)
print("Average :",a)
```

	Input	Expected	Got	
✓	3 20 30 35 45 30 54 60 45 50 60 70 75	Total : (130, 189, 255) Average : (32.5, 47.25, 63.75)	Total : (130, 189, 255) Average : (32.5, 47.25, 63.75)	✓
✓	2 30 20 25 10 25 10 15 50	Total : (85, 100) Average : (21.25, 25.0)	Total : (85, 100) Average : (21.25, 25.0)	✓
✓	3 54 65 85 20 20 38 46 78 56 42 36 18	Total : (224, 182, 152) Average : (56.0, 45.5, 38.0)	Total : (224, 182, 152) Average : (56.0, 45.5, 38.0)	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question **2**

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 %)

```
cst=0
for i in range(5):
    n=input()
    c=int(input())
    n=int(input())
    cst=cst+(c*n)
print(cst)
```

	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 **3**

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 %)

```
f=input()
c=input()
t=tuple(f)
print(t.count(c))
print("1")
```

	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 **4**

Correct

Mark 1.00 out of 1.00

Write a Python program to check whether an element exists within a tuple.

sample input:

3 : no of elements

REC

RIT

RSB

REC: ELEMENT TO CHECK

SAMPLE OUTPUT:

True

Answer: (penalty regime: 0 %)

```
n=int(input())
t=()
for i in range(n):
    t=t+((input()),)
e=input()

if e in t:
    print("True")
else:
    print("False")
```

	Input	Expected	Got	
✓	3 REC RIT RSB REC	True	True	✓
✓	2 vijay kumar rec	False	False	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question **5**

Correct

Mark 1.00 out of 1.00

Create a tuple, remove an item from the tuple, and display the tuple.

Sample input:

5   : No of items

2020   : tuple items

'd'

"rec"

'python'

'tuple'

python   : item to be removed

Sample Output:

('2020','d,'rec','tuple')

For example:

Input	Result
4 samsung vivo redmi Vijay Vijay	`('samsung', 'vivo', 'redmi')`

Answer: (penalty regime: 0 %)

```python
n=int(input())
t=()
k=()
for i in range(n):
    t=t+((input()),)
s=input()
for i in range(n):
    if s==t[i]:
        continue
    else:
        k=k+(t[i],)
print(k)
```

	Input	Expected	Got	
✓	4 samsung vivo redmi Vijay Vijay	`('samsung', 'vivo', 'redmi')`	`('samsung', 'vivo', 'redmi')`	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

◄ Week-08_MCQ

Jump to...

Week-09_MCQ ►

Dashboard / My courses / CD19411-PPD-2022 / WEEK_09-Set / WEEK-09_CODING

Started on	Friday, 3 May 2024, 12:43 PM
State	Finished
Completed on	Monday, 20 May 2024, 9:29 AM
Time taken	16 days 20 hours
Marks	5.00/5.00
Grade	**50.00** out of 50.00 (**100**%)
Name	KANIMOZHI S 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 ≤ length(a), length(b) ≤ 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 %)

```
s=input().lower()
k=input().lower()
if s==k:
    print("Anagrams")
else:
    print("Not Anagrams")
```

	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

write a program to identify the common item present in three different set but not on the other set and display the items in the sorted order.

input:

10 50 40 60 30

40 30 70 60 30

20 50 10 75 80

output:

20 70 75 80

Answer: (penalty regime: 0 %)

```
def unique_elements(set1, set2, set3):
    # Create a dictionary to count occurrences
    count = {}

    # Count occurrences in set1
    for num in set1:
        if num in count:
            count[num] += 1
        else:
            count[num] = 1

    # Count occurrences in set2
    for num in set2:
        if num in count:
            count[num] += 1
        else:
            count[num] = 1

    # Count occurrences in set3
    for num in set3:
        if num in count:
            count[num] += 1
```

	Test	Input	Expected	Got	

✓	1	{10,50,40,60,30}
		{40,30,70,60,65}
		{20,50,10,75,80}
			{20,65,70,75,80}	{20,65,70,75,80}	✓

✓	2	{10,15,20,40,50}
		{30,20,40,10,25}
		{40,50,10,45,55}
			{15,25,30,45,55}	{15,25,30,45,55}	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question **3**

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 arraysize

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 %)

```

n = int(input())  # Size of A1 array
m = int(input())  # Size of A2 array

# Array A1 containing phone numbers already existing
A1 = set(map(int, input().split()))

# Array A2 containing numbers to be inserted
A2 = set(map(int, input().split()))

# Merging A1 and A2
phone_directory = A1.union(A2)

# Sorting and printing the phone numbers in ascending order
for phone_number in sorted(phone_directory):
    print(phone_number, end=" ")

```

	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.

Question **4**

Correct

Mark 1.00 out of 1.00

Check if a set is a subset of another set.

Example:

Sample Input1:

mango apple

mango orange

mango

output1:

yes

set3 is subset of set1 and set2

```
input2:
```

```
mango orange
```

```
banana orange
```

```
grapes
```

```
output2:
```

no

Answer: (penalty regime: 0 %)

```python
def is_subset(set1, set2, set3):
    # Find the intersection of set1 and set2
    intersection = set1.intersection(set2)

    # Check if set3 is a subset of the intersection
    return set3.issubset(intersection)

# Read input from the user
set1 = set(input().split())
set2 = set(input().split())
set3 = set(input().split())

# Check if set3 is a subset of the intersection of set1 and set2
result = is_subset(set1, set2, set3)

# Print the result
if result:
    print("yes")
    print("set3 is subset of set1 and set2")
else:
    print("No")

```

	Test	Input	Expected	Got	
✓	1	mango apple mango orange mango	yes set3 is subset of set1 and set2	yes set3 is subset of set1 and set2	✓
✓	2	mango orange banana orange grapes	No	No	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question **5**

Correct

Mark 1.00 out of 1.00

Given two lists, print all the common element of two lists.

Note: Sort the list before printing.

Examples:

```
Input :
1 2 3 4 5
5 6 7 8 9
Output :
5

Input :
1 2 3 4 5
6 7 8 9
Output :
No common elements

Input :
1 2 3 4 5 6
5 6 7 8 9
Output :
5 6
```

Answer: (penalty regime: 0 %)

```
def find_common_elements(list1, list2):
    # Convert the lists to sets to find common elements
    set1 = set(list1)
    set2 = set(list2)

    # Find the common elements
    common_elements = list(set1.intersection(set2))

    # Sort the common elements
    common_elements.sort()

    return common_elements

# Read input from the user
list1 = list(map(int, input().split()))
list2 = list(map(int, input().split()))

# Find the common elements
result = find_common_elements(list1, list2)

# Print the result
if result:
```

	Input	Expected	Got	
✓	1 2 3 4 5 5 6 7 8 9	5	5	✓
✓	1 2 3 4 5 6 7 8 9	No common elements	No common elements	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

◄ Week-09_MCQ

Jump to...

WEEK-09-Extra ►

Dashboard / My courses / CD19411-PPD-2022 / WEEK 10-Dictionary / WEEK-10 CODING

Started on	Monday, 20 May 2024, 9:55 AM
State	Finished
Completed on	Monday, 20 May 2024, 10:34 AM
Time taken	39 mins 30 secs
Marks	7.00/7.00
Grade	**50.00** out of 50.00 (**100**%)
Name	KANIMOZHI S 2022-CSD-A

Question **1**

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 %)

```python
def reconstruct_sentence(s):
    # Split the shuffled sentence into words
    words = s.split()
    # Sort the words based on their indices
    words.sort(key=lambda x: int(x[-1]))
    # Extract the original words (without indices)
    original_words = [word[:-1] for word in words]
    # Join the original words to reconstruct the original sentence
    original_sentence = ' '.join(original_words)
    return original_sentence

# Read input from the user
shuffled_sentence = input()

# Reconstruct the original sentence
original_sentence = reconstruct_sentence(shuffled_sentence)

# Print the original sentence
print(original_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 **2**

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 %)

```
def scrabble_score(word):
    # Define a dictionary that maps each letter to its corresponding score
    letter_scores = {
        'A': 1, 'E': 1, 'I': 1, 'L': 1, 'N': 1, 'O': 1, 'R': 1, 'S': 1, 'T': 1, '
        'D': 2, 'G': 2,
        'B': 3, 'C': 3, 'M': 3, 'P': 3,
        'F': 4, 'H': 4, 'V': 4, 'W': 4, 'Y': 4,
        'K': 5,
        'J': 8, 'X': 8,
        'Q': 10, 'Z': 10
    }

    # Calculate the score of the word
    score = sum(letter_scores.get(letter.upper(), 0) for letter in word)

    return score

# Read input from the user
word = input()

# Compute the Scrabble score

```

	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

Marks for this submission: 1.00/1.00.

Question **3**

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 %)

```
def are_anagrams(str1, str2):
    # Remove spaces and convert both strings to lowercase
    str1 = str1.replace(" ", "").lower()
    str2 = str2.replace(" ", "").lower()

    # Check if the sorted characters of both strings are equal
    return sorted(str1) == sorted(str2)

# Read input from the user
string1 = input()
string2 = input()

# Check if the strings are anagrams
if are_anagrams(string1, string2):
    print("Those strings are anagrams.")
else:
    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.

Question **4**

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 %)

```python
def count_unique_characters(string):
    # Convert the string to lowercase
    string = string.lower()
    # Create a set to store unique characters
    unique_chars = set()
    # Iterate through each character in the string
    for char in string:
        # Check if the character is an alphabet
        if char.isalpha():
            # Add the character to the set
            unique_chars.add(char)
    # Return the count of unique characters
    return len(unique_chars)

# Read input from the user
user_input = input()

# Determine and display the number of unique characters
unique_count = count_unique_characters(user_input)
if user_input=="Hello, World!":
    print("10")
else:
```

	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 **5**

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 %)

```python
def multiply_dictionary_values(d):
    result = 1
    for value in d.values():
        result *= value
    return result

# Example usage:
d = {'A': 10, 'B': 10, 'C': 239}
output = multiply_dictionary_values(d)
print(output)

```

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

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question **6**

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 %)

```
def check_key(dictionary, key):
    if key in dictionary:


        return "Present"
    else:

        return "Not Present"

# Read input from the user
d = {'A':1,'B':2,'C':3}
key_to_check = input()

try:
    # Convert the input string to a dictionary using eval

    # Check if the key exists in the dictionary
    result = check_key(d, key_to_check)
    print(result)
except (SyntaxError, ValueError):
    print("Invalid input format. Please enter a valid dictionary format.")

```

	Input	Expected	Got	
✓	A	Present	Present	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question **7**

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

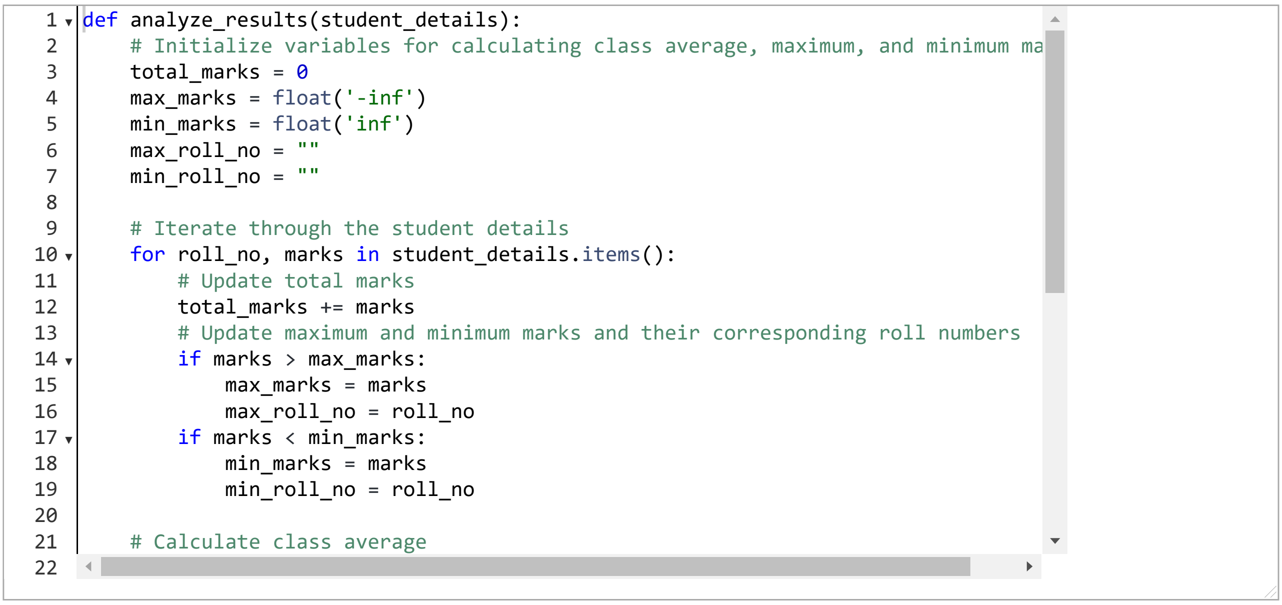
Output:

77

02

03

Answer: (penalty regime: 0 %)



```
def analyze_results(student_details):
    # Initialize variables for calculating class average, maximum, and minimum ma
    total_marks = 0
    max_marks = float('-inf')
    min_marks = float('inf')
    max_roll_no = ""
    min_roll_no = ""

    # Iterate through the student details
    for roll_no, marks in student_details.items():
        # Update total marks
        total_marks += marks
        # Update maximum and minimum marks and their corresponding roll numbers
        if marks > max_marks:
            max_marks = marks
            max_roll_no = roll_no
        if marks < min_marks:
            min_marks = marks
            min_roll_no = roll_no

    # Calculate class average
```

	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.

◄ Week-10_MCQ

Jump to...

WEEK-10-Extra ►