Week 1

Started on	Wednesday, 28 February 2024, 10:22 AM
State	Finished
Completed on	Wednesday, 28 February 2024, 11:42 AM
Time taken	1 hour 20 mins
Marks	5.00/5.00
Grade	**50.00** out of 50.00 (**100**%)
Name	CHARAN RAJ D K 2022-CSD-A

Question **1**

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 %)

```
x=input()
y=input()
z=float(y)
print(x,type(int(x)),sep=",")
print("%.1f"%z,type(z),sep=",")
```

	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 **2**

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())
a=z-(x+y)
b=(a/(x+y))*100
print("{0:.2f} is the gain percent.".format(b))
```

	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.	✔
✔	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 **3**

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 %)

```
salary=int(input())
x=abs((salary-500)/130)
y=x+10;
print("weekdays {0:.2f}".format(y))
print("weekend {0:.2f}".format(x))
```

	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	✔
✔	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 **4**

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
x=float(input())
print("{0:.3f}".format(math.sqrt(x)))
```

	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.

Question **5**

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 %)

```
x=int(input())
y=int(input())
a=x*0.10
b=y*0.25
r=a+b
print("Your total refund will be ${0:.2f}.".format(r))
```

	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.

Week 2

Started on	Tuesday, 5 March 2024, 8:07 AM
State	Finished
Completed on	Tuesday, 5 March 2024, 8:26 AM
Time taken	18 mins 55 secs
Marks	5.00/5.00
Grade	**50.00** out of 50.00 (**100**%)
Name	CHARAN RAJ D K 2022-CSD-A

Question **1**

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 %)

```
x=int(input())
y=int(input())
print(x>=18 and y>40)
```

	Input	Expected	Got	
✔	19 45	True	True	✔

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

Question **2**

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 %)

```
x=int(input())
y=int(input())
x=x*75
y=y*112
z=x+y
print("The total weight of all these widgets and gizmos is {0} grams.".format(z))
```

	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.

Question **3**

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())
p1=int(input())
p2=int(input())
p3=int(input())
p4=int(input())
print(p1%n==0,p2%n==0,p3%n==0,p4%n==0)
```

	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 **4**

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 %)

```
x=int(input())
y=int(input())
if(x%3==0 and y%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 **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 %)

```
x=int(input())
i=1
while(i<4):
    y=x*(4/100)
    x=y+x
    print("Balance as of end of Year {0}: ${1:.2f}.".format(i,x))
    i=i+1
```

	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 3

Started on	Tuesday, 5 March 2024, 9:03 AM
State	Finished
Completed on	Tuesday, 5 March 2024, 9:46 AM
Time taken	42 mins 51 secs
Marks	5.00/5.00
Grade	**50.00** out of 50.00 (**100**%)
Name	CHARAN RAJ D K 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 %)

```
x=int(input())
if(x%2==0):
    print("{0} is even.".format(x))
else:
    print("{0} is odd.".format(x))
```

	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

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 %)

```
x=float(input())
d=100
if(x<84):
    print("{0:.2f}".format(d))
if(x>84 and x<=199):
    print("{0:.2f}".format(x*1.20))
if(x>199 and x<400):
    print("{0:.2f}".format(x*1.50))
if(x>400 and x<600):
    print("{0:.2f}".format((x*1.80)+((x*1.80)*0.15)))
if(x>600):
    print("{0:.2f}".format((x*2.00)+((x*2.00)*0.15)))
```

	Input	Expected	Got	
✔	50	100.00	100.00	✔
✔	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 **3**

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 Input3

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 %)

```
x=input()
if(x=="a" or x=="e" or x=="i" or x=="o" or x=="u"):
    print("It's a vowel.")
elif(x=="y" or x=="h"):
    print("Sometimes it's a vowel... Sometimes it's a consonant.")
else:
    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.	✔
✔	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 **4**

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 %)

```
x=int(input())
y=input()
z=int(input())
if(y=="+"):
    print(x+z)
if(y=="-"):
    print(x-z)
if(y=="*"):
    print(x*z)
if(y=="/"):
    print(x/z)
```

	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 **5**

Correct

Mark 1.00 out of 1.00

The length of a month varies from 28 to 31 days. In this exercise you will create a program that reads the name of a month from the user as a string. Then your program should display the number of days in that month. Display "28 or 29 days" for February so that leap years are addressed.

Sample Input 1

February

Sample Output 1

February has 28 or 29 days in it.

Sample Input 2

March

Sample Output 2

March has 31 days in it.

Sample Input 3

April

Sample Output 3

April has 30 days in it.

For example:

Input	Result
February	February has 28 or 29 days in it.

Answer: (penalty regime: 0 %)

```
x=input()
if(x=="January" or x=="March" or x=="May" or x=="July" or x=="August" or x=="October"
    print("{0} has 31 days in it.".format(x))
elif(x=="February"):
    print("{0} has 28 or 29 days in it.".format(x))
else:
    print("{0} has 30 days in it.".format(x))
```

	Input	Expected	Got	
✔	February	February has 28 or 29 days in it.	February has 28 or 29 days in it.	✔
✔	March	March has 31 days in it.	March has 31 days in it.	✔
✔	April	April has 30 days in it.	April has 30 days in it.	✔
✔	May	May has 31 days in it.	May has 31 days in it.	✔

Passed all tests! ✔

Week 4

Started on	Thursday, 25 April 2024, 9:20 AM
State	Finished
Completed on	Thursday, 25 April 2024, 9:52 AM
Time taken	32 mins 11 secs
Marks	5.00/5.00
Grade	**50.00** out of 50.00 (**100**%)
Name	CHARAN RAJ D K 2022-CSD-A

Question **1**

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 %)

```
def fibonacci(n):
  if n < 0:
    return -1
  elif n == 0:
    return 0
  elif n == 1:
    return 1
  else:
    return fibonacci(n-1) + fibonacci(n-2)
n = int(input())
result = fibonacci(n-1)
print(result)
```

	Input	Expected	Got	
✔	4	2	2	✔
✔	8	13	13	✔

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 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 %)

```
def count_digits(number):
    count = 0
    while number > 0:
        count += 1
        number //= 10
    return count
num = int(input())
print(count_digits(num))
```

	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 **3**

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 %)

```
def compute_average():
    total = 0
    count = 0
    value = float(input())
    if value == 0:
        print("Error: The first value cannot be 0.")
        return

    while value != 0:
        total += value
        count += 1
        value = float(input())

    if count > 0:
        average = total / count
        print("The average is {0}.".format(average))
compute_average()
```

	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 **4**

Correct

Mark 1.00 out of 1.00

Write a program to find the sum of the series 1 +11 + 111 + 1111 + . . . + 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 %)

```
def series_sum(n):
    sum_of_series = 0
    term = 0
    for i in range(1, n + 1):
        term = term * 10 + 1
        sum_of_series += term
    return sum_of_series
n = int(input())
print(series_sum(n))
```

	Input	Expected	Got	
✔	1	1	1	✔
✔	3	123	123	✔
✔	4	1234	1234	✔
✔	7	1234567	1234567	✔

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

Question **5**

Correct

Mark 1.00 out of 1.00

A number is stable if each digit occur the same number of times.i.e, the frequency of each digit in the number is the same. For e.g. 2277,4004,11,23,583835,1010 are examples for stable numbers.
Similarly, a number is unstable if the frequency of each digit in the number is NOT same.

Sample Input:

2277

Sample Output:

Stable Number

Sample Input 2:

121

Sample Output 2:

Unstable Number

For example:

Input	Result
2277	Stable Number

Answer: (penalty regime: 0 %)

```
from collections import Counter
def is_stable_number(number):
    digit_counts = Counter(str(number))
    return len(set(digit_counts.values())) == 1
number = input()
if is_stable_number(number):
    print("Stable Number")
else:
    print("Unstable Number")
```

	Input	Expected	Got	
✔	9988	Stable Number	Stable Number	✔
✔	2277	Stable Number	Stable Number	✔
✔	1233	Unstable Number	Unstable Number	✔

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

Week 5

Started on	Wednesday, 3 April 2024, 10:50 AM
State	Finished
Completed on	Wednesday, 3 April 2024, 11:37 AM
Time taken	47 mins 7 secs
Marks	5.00/5.00
Grade	**50.00** out of 50.00 (**100**%)
Name	CHARAN RAJ D K 2022-CSD-A

Question **1**

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 5 10 -15 -20 -99	-15 -20 0 5 10

Answer: (penalty regime: 0 %)

```
def group():
  negative = []
  zeros = []
  positive = []
  while True:
    number = int(input())
    if number == -99:
      break
    elif number < 0:
      negative.append(number)
    elif number == 0:
      zeros.append(number)
    else:
      positive.append(number)
  return negative, zeros, positive
def grouped(negative, zero, positive):
  print(*negative, sep="\n")
  print(*zero, sep="\n")
  print(*positive, sep="\n")
```

```
20 negative, zeros, positive= group()
21 grouped(negative, zeros, positive)
```

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

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

Question **2**

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 %)

```
def print_distinct(arr):
    unique_elements = set(arr)
    print(*unique_elements)
n = int(input())
arr= [int(input()) for _ in range(n)]
print_distinct(arr)
```

	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 **3**

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 %)

```
def common(l1, l2):
    s1 = set(l1)
    s2 = set(l2)
    return len(s1.intersection(s2)) > 0
l1 = input().split()
l2 = input().split()
l1 = list(map(int, l1))
l2 = list(map(int, l2))
print(common(l1, l2))
```

	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 **4**

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 11 22 55 44 0	55 44 33 22 11

Answer: (penalty regime: 0 %)

```
numbers = []
while True:
    num = int(input())
    if num == 0:
        break
    numbers.append(num)
numbers.sort(reverse=True)
for num in numbers:
    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 **5**

Correct

Mark 1.00 out of 1.00

You have to generate the sum of specific numbers based on its position in the array set provided to you.

This is explained below:

Example 1:

Let us assume the encoded set of numbers given to you is:

input1: 5

input2: {1, 51, 436, 7860, 41236}

Step 1:

Starting from the 0

index of the array pick up digits as per below:

0 index – pick up the units value of the number (in this case is 1).

1 index - pick up the tens value of the number (in this case it is 5).

2 index - pick up the hundreds value of the number (in this case it is 4).

3 index - pick up the thousands value of the number (in this case it is 7).

4 index - pick up the ten thousands value of the number (in this case it is 4).

(Continue this for all the elements of the input array).

The array generated from Step 1 will then be – {1, 5, 4, 7, 4}.

Step 2:

Square each number present in the array generated in Step 1.

{1, 25, 16, 49, 16}

Step 3:

Calculate the sum of all elements of the array generated in Step 2 to get the final result. The result will be = 107.

Note:

Note:

1) While picking up a number in Step1, if you observe that the number is smaller than the required position then use 0.

2) input1 represents the number of elements in input2 and

input2[] is the array of numbers.

For example:

Input	Result
5 1 51 436 7860 41236	107

Answer: (penalty regime: 0 %)

```
def sum_num(n, arr):
    step1 = [int(str(arr[i])[-(i+1)]) if len(str(arr[i])) > i else 0 for i in range(n)]
    step2 = [i**2 for i in step1]
    result = sum(step2)
    return result
n = int(input())
arr= [int(input()) for _ in range(n)]
print(sum_num(n,arr))
```

	Input	Expected	Got	
✔	5 1 51 436 7860 41236	107	107	✔
✔	4 1 1 111 1111	3	3	✔

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

Week 6

Started on	Wednesday, 3 April 2024, 10:32 AM
State	Finished
Completed on	Friday, 5 April 2024, 12:16 PM
Time taken	2 days 1 hour
Marks	5.00/5.00
Grade	**50.00** out of 50.00 (**100**%)
Name	CHARAN RAJ D K 2022-CSD-A

Question **1**

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', '
input_word = input()
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 **2**

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 %)

```
def reverse_string(s):
    s_list = list(s)
    left, right = 0, len(s_list) - 1
    while left < right:
        s_list[left], s_list[right] = s_list[right], s_list[left]
        left += 1
        right -= 1
    reversed_string = ''.join(s_list)
    return reversed_string
input_string = input()
reversed_string = reverse_string(input_string)
print(reversed_string)

```

	Input	Expected	Got	
✔	hello	olleh	olleh	✔
✔	Hannah	hannaH	hannaH	✔

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

Question **3**

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 %)

```
s=input()
def length(s):
    words = s.split()
    if len(words) == 0:
        return 0
    return len(words[-1])
print(length(s))
```

	Input	Expected	Got	
✔	Hello World	5	5	✔

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

Question **4**

Correct

Mark 1.00 out of 1.00

Given a string, determine if it is a palindrome, considering only alphanumeric characters and ignoring cases.

Note: For the purpose of this problem, we define empty string as valid palindrome.

Example 1:

```
Input:
A man, a plan, a canal: Panama

Output:
1
```

Example 2:

```
Input:
race a car

Output:
0
```

Constraints:

- `s` consists only of printable ASCII characters.

Answer: (penalty regime: 0 %)

```
s=input()
def isPalindrome(s):
    filtered_s = ''.join(char.lower() for char in s if char.isalnum())
    return filtered_s == filtered_s[::-1]
if(isPalindrome(s)):
    print("1")
else:
    print("0")

```

	Input	Expected	Got	
✔	A man, a plan, a canal: Panama	1	1	✔
✔	race a car	0	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 challenge, the user enters a string and a substring. You have to print the number of times that the substring occurs in the given string. String traversal will take place from left to right, not from right to left.

NOTE: String letters are case-sensitive.

Input Format

The first line of input contains the original string. The next line contains the substring.

Constraints

1<=len(string)<=200

Each character in the string is an *ascii* character.

Output Format

Output the integer number indicating the total number of occurrences of the substring in the original string.

For example:

Input	Result
ABCDCDC CDC	2

Answer: (penalty regime: 0 %)

```
def count_substring(string, sub_string):
    count = 0
    for i in range(len(string) - len(sub_string) + 1):
        if string[i:i + len(sub_string)] == sub_string:
            count += 1
    return count
string = input().strip()
sub_string = input().strip()
print(count_substring(string, sub_string))
```

	Input	Expected	Got	
✔	ABCDCDC CDC	2	2	✔
✔	XYMAMAM MAM	2	2	✔

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

Week 7

Started on	Thursday, 9 May 2024, 9:37 PM
State	Finished
Completed on	Thursday, 9 May 2024, 10:26 PM
Time taken	48 mins 34 secs
Marks	5.00/5.00
Grade	**50.00** out of 50.00 (**100**%)
Name	CHARAN RAJ D K 2022-CSD-A

Question **1**

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

```
def sumofsquares(m):
  if m <= 0:
    return False
  i = 1
  while i * i <= m:
    remaining = m - i * i
    if int(remaining**0.5) == remaining**0.5:
      return True
    i += 1
  return False
```

	Test	Expected	Got	
✔	print(sumofsquares(41))	True	True	✔
✔	print(sumofsquares(30))	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 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):
    numbers = sorted([a, b, c])
    return numbers[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 **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

```
def isPrime(num):
  if num <= 1:
    return False
  if num % 2 == 0:
    return num == 2
  i = 3
  while i * i <= num:
    if num % i == 0:
      return False
    i += 2
  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 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

```
def read_and_total():
  line = input()
  if not line:
    return 0
  try:
    return int(line) + read_and_total()
  except ValueError:
    return read_and_total()
total = read_and_total()
print(total)
```

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

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 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

```
def checkPassword (input1):
    if len(input1) < 8:
        print("That isn't a good password.")
        return
    has_upper = False
    has_lower = False
    has_digit = False
    for char in input1:
        if char.isupper():
            has_upper= True
        elif char.islower():
            has_lower =True
        elif char.isdigit():
            has_digit= True
    if has_upper and has_lower and has_digit:
        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.

Week 8

Started on	Sunday, 19 May 2024, 5:14 PM
State	Finished
Completed on	Sunday, 19 May 2024, 5:35 PM
Time taken	21 mins 35 secs
Marks	5.00/5.00
Grade	**50.00** out of 50.00 (**100**%)
Name	CHARAN RAJ D K 2022-CSD-A

Question **1**

Correct

Mark 1.00 out of 1.00

Create a tuple t1 with numbers 1 to 5, t2 with 6 to 10 and t3 with a string "REC".

Concatenate t1 and t2 and print the result.

Repeat the t3 10 times without using any looping statements.

Expected output:

(1, 2, 3, 4, 5, 6, 7, 8, 9, 10)
('REC', 'REC', 'REC', 'REC', 'REC', 'REC', 'REC', 'REC', 'REC', 'REC')

Answer: (penalty regime: 0 %)

```
t1 = (1, 2, 3, 4, 5)
t2 = (6, 7, 8, 9, 10)
t3 = ("REC",)
t1_t2 = t1 + t2
print(t1_t2)
t3_repeated = t3 * 10
print(t3_repeated)
```

	Expected	Got	
✔	(1, 2, 3, 4, 5, 6, 7, 8, 9, 10) ('REC', 'REC', 'REC', 'REC', 'REC', 'REC', 'REC', 'REC', 'REC', 'REC')	(1, 2, 3, 4, 5, 6, 7, 8, 9, 10) ('REC', 'REC', 'REC', 'REC', 'REC', 'REC', 'REC', 'REC', 'REC', 'REC')	✔

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 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 %)

```
num_elements = int(input())
elements = []
for i in range(num_elements):
    element = input()
    elements.append(element)
elements = tuple(elements)
element_to_check = input()
exists = element_to_check in elements
print(exists)
```

	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 **3**

Correct

Mark 1.00 out of 1.00

Create a tuple:

my_tuple = ('R','a','j','a','l','a','k','s','h','m','i')

and apply slicing and display the output as shown below:

('R', 'a', 'j', 'a')
('l', 'a', 'k', 's', 'h', 'm', 'i')
('R', 'a', 'j')
('l', 'a', 'k')
('m', 'i')

Answer: (penalty regime: 0 %)

```
my_tuple = ('R','a','j','a','l','a','k','s','h','m','i')
print(my_tuple[:4])
print(my_tuple[4:])
print(my_tuple[:3])
print(my_tuple[4:7])
print(my_tuple[-2:])
```

	Expected	Got	
✔	('R', 'a', 'j', 'a') ('l', 'a', 'k', 's', 'h', 'm', 'i') ('R', 'a', 'j') ('l', 'a', 'k') ('m', 'i')	('R', 'a', 'j', 'a') ('l', 'a', 'k', 's', 'h', 'm', 'i') ('R', 'a', 'j') ('l', 'a', 'k') ('m', 'i')	✔

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 unpack the following tuple into variables depends on the length of tuple (Max length = 10) and display each values separately.

Sample Input:

4

10

30

40

60

Sample Output:

a=10

b=30

c=40

d=60

Answer: (penalty regime: 0 %)

```
n=int(input())
List1=['a','b','c','d','e','f','g','h','i','j','k','l','m','n','o','p','q','r','s','t','u','v
List=[]
for i in range(n):
    List.append(int(input()))
for i in range(len(List)):
    print("%s=%d"%(List1[i],List[i]))
```

	Input	Expected	Got	
✔	4 10 30 40 60	a=10 b=30 c=40 d=60	a=10 b=30 c=40 d=60	✔
✔	9 15 60 75 85 90 70 35 25 45	a=15 b=60 c=75 d=85 e=90 f=70 g=35 h=25 i=45	a=15 b=60 c=75 d=85 e=90 f=70 g=35 h=25 i=45	✔

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 %)

```
n = int(input())
items = []
for _ in range(n):
    items.append(input())
item_to_remove = input()
my_tuple = tuple(items)
my_tuple = tuple(item for item in my_tuple if item != item_to_remove)
print(my_tuple)

```

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

Passed all tests! ✔

Week 9

Started on	Sunday, 19 May 2024, 5:35 PM
State	Finished
Completed on	Tuesday, 21 May 2024, 10:41 PM
Time taken	2 days 5 hours
Marks	5.00/5.00
Grade	**50.00** out of 50.00 (**100**%)
Name	CHARAN RAJ D K 2022-CSD-A

Question **1**

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 %)

```
def window(win):
    return len(set(win))
List=list(map(int,input().split(" ")))
k=int(input())
for i in range(len(List)-k+1):
    print(window(List[i:k+i]))
```

	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 **2**

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 %)

```
sentence = input()
words = sentence.split()
unique_sorted_words = sorted(set(words))
count = sum(1 for word in unique_sorted_words if len(word) > 3)
print(" ".join(unique_sorted_words))
print("Count =", 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 **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())
m=int(input())
List1=list(map(int,input().split(" ")))
List2=list(map(int,input().split(" ")))
for i in List2:
    List1.append(i)
List1=list(set(List1))
List1.sort()
for i in List1:
    print(i,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! ✔

Question **4**

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 %)

```
List1=list(map(int,input().split(" ")))
List2=list(map(int,input().split(" ")))
List=list(set(List1) & set(List2))
if List:
    for i in List:
        print(i,end=" ")
else:
    print("No common elements")
```

	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.

Question **5**

Correct

Mark 1.00 out of 1.00

A number is stable if each digit occur the same number of times.i.e, the frequency of each digit in the number is the same. For e.g. 2277,4004,11,23,583835,1010 are examples for stable numbers.
Similarly, a number is unstable if the frequency of each digit in the number is NOT same.

Sample Input:

2277

Sample Output:

Stable Number

Sample Input 2:

121

Sample Output 2:

Unstable Number

Answer: (penalty regime: 0 %)

```
num = input()
digit_freq = {}
for digit in num:
    if digit.isdigit():
        digit_freq[digit] = digit_freq.get(digit, 0) + 1
if len(set(digit_freq.values())) == 1:
    print("Stable Number")
else:
    print("Unstable Number")
```

	Input	Expected	Got	
✔	9988	Stable Number	Stable Number	✔
✔	12	Stable Number	Stable Number	✔
✔	455	Unstable Number	Unstable Number	✔

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

Week 10

Started on	Sunday, 19 May 2024, 5:40 PM
State	Finished
Completed on	Tuesday, 21 May 2024, 10:42 PM
Time taken	2 days 5 hours
Marks	7.00/7.00
Grade	**50.00** out of 50.00 (**100**%)
Name	CHARAN RAJ D K 2022-CSD-A

Question **1**

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 %)

```
w1 = input().lower()
w2 = input().lower()
word1 = ''.join(sorted(w1))
word2 = ''.join(sorted(w2))
if word1 == word2:
    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 **2**

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 %)

```
n = int(input())
total_marks = 0
max_marks = -1
min_marks = float('inf')
max_roll_no = ""
min_roll_no = ""
for i in range(n):
    roll_no, marks = input().split()
    marks = int(marks)
    total_marks += marks
    if marks > max_marks:
        max_marks = marks
        max_roll_no = roll_no
    if marks < min_marks:
        min_marks = marks
        min_roll_no = roll_no
class_avg = total_marks // n
print(class_avg)
print(max_roll_no)
print(min_roll_no)
```

	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 **3**

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 %)

```
s = input().split()
def pos(word):
    return int(word[-1])
s.sort(key=pos)
o = ' '.join(word[:-1] for word in s)
print(o)
```

	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 **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 %)

```
x = input()
y = set(x)
c = x.count(' ')
special = sum(1 for char in x if not char.isalnum() and not char.isspace())
print(len(y))
```

	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

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 %)

```
d = {'A':1, 'B':2, 'C':3}
a = input()
if a in d:
    print("Present")
else:
    print("Not Present")
```

	Input	Expected	Got	
✔	A	Present	Present	✔

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

Question **6**

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 %)

```
a = {'a': 10, 'b': 10, 'c': 239}
result = 1
for value in a.values():
    result *= value
print(result)
```

	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 **7**

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 %)

```
x = {'A': 1, 'E': 1, 'I': 1, 'L': 1, 'N': 1, 'O': 1, 'R': 1, 'S': 1, 'T': 1, 'U': 1,'D': 2, '
word = input().upper()
score = sum(x.get(letter, 0) for letter in word)
print("{0} is worth {1} points.".format(word, 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.