

RAJALAKSHMI ENGINEERING COLLEGE
RAJALAKSHMI NAGAR, THANDALAM – 602 105



RAJALAKSHMI
ENGINEERING COLLEGE
An AUTONOMOUS Institution
Affiliated to ANNA UNIVERSITY, Chennai

Laboratory Record Notebook

Name :

Year / Branch / Section :

Register No. :

College Roll No. :

Semester :

Academic Year :



RAJALAKSHMI ENGINEERING COLLEGE

An AUTONOMOUS institution
Affiliated to ANNA UNIVERSITY, Chennai

BONAFIDE CERTIFICATE

NAME :

ACADEMIC YEAR SEMESTER BRANCH

REGISTER NO.

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

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

Signature of Faculty - in - Charge

Submitted for the Practical Examination held on

Internal Examiner

External Examiner

INDEX

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

Started on	Wednesday, 28 February 2024, 10:21 AM
State	Finished
Completed on	Wednesday, 28 February 2024, 11:39 AM
Time taken	1 hour 17 mins
Marks	5.00/5.00
Grade	50.00 out of 50.00 (100%)
Name	KEERTHI AACHUTHAN K 2022-CSD-A

Question 1

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 %)

```
1 import math
2 x=float(input())
3 print("{:.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 2

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 %)

```
1 | a=int(input())
2 | b=abs((a-500)/130)
3 | print("weekdays {0:.2f}\nweekend {1:.2f}".format((10+b),(b)))
```

	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 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	Your total refund will be \$7.00.
20	

Answer: (penalty regime: 0 %)

```

1 | a=int(input())
2 | b=int(input())
3 |
4 | x=a*0.10+b*0.25
5 | print("Your total refund will be ${:.2f}.".format(x))
  
```

	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	46.34 is the gain percent.
250	
15000	

Answer: (penalty regime: 0 %)

```

1 x=int(input())
2 y=int(input())
3 z=int(input())
4
5 cost=x+y
6 gain=z-cost
7 percent=(gain/cost)*100
8 print("{0:.2f} is the gain percent.".format(percent))

```

	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 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 %)

```
1 a=int(input())
2 b=float(input())
3 print(str(a)+", "+str(type(a)))
4 print(str("{0:.1f}".format(b))+", "+str(type(b)))
```

	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.

[◀ Week-1_MCQ](#)

Jump to...

[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, 8:53 AM

Time taken 43 mins 41 secs

Marks 5.00/5.00

Grade **50.00** out of 50.00 (**100%**)

Name [KEERTHI AACHUTHAN K 2022-CSD-A](#)

Question 1

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

Answer: (penalty regime: 0 %)

```

1
2 x= int(input())
3 y= int(input())
4
5 totalweight = (x *75) + (y *112)
6
7 print("The total weight of all these widgets and gizmos is {} grams.".format(totalwei
8

```

	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 2

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 %)

```

1 n=int(input())
2 for i in range (1,4):
3     interest= 0.04 * n
4     n=n+interest
5     print("Balance as of end of Year %d: $%.2f."%(i,n))
6

```

	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.

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, $0 \leq x \leq 1$.**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 %)

```

1 | x=int(input())
2 | value=ord('C')+x
3 | output =chr(value)
4 | print(output)
5 |

```

	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

In London, every year during Dasara there will be a very grand doll show. People try to invent new dolls of different varieties. The best-sold doll's creator will be awarded with a cash prize. So people broke their heads to create dolls innovatively. Knowing this competition, Mr.Lokpaul tried to create a doll that sings only when an even number is pressed and the number should not be zero and greater than 100.

IF Lokpaul wins print true, otherwise false.

Sample Input

10

Sample Output

True

Explanation:

Since 10 is an even number and a number between 0 and 100, True is printed

Answer: (penalty regime: 0 %)

```

1 | x=int(input())
2 |
3 v if (1<=x<=100 and x%2==0 and x!=0):
4 |   print("True")
5 v else:
6 |   print("False")
7

```

	Input	Expected	Got	
✓	56	True	True	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question 5

Correct

Mark 1.00 out of 1.00

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

Input format:

Line 1 has the total number of weapons

Line 2 has the total number of Soldiers.

Output Format:

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

Sample Input:

32

43

Sample Output:'

False

Answer: (penalty regime: 0 %)

```

1 |x=int(input())
2 |y=int(input())
3 |if(x%3==0 and y%2==0):
4 |    print("True")
5 |else:
6 |    print("False")
7 |

```

	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.

◀ 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 Monday, 22 April 2024, 8:35 PM

State Finished

Completed on Wednesday, 24 April 2024, 8:57 PM

Time taken 2 days

Marks 5.00/5.00

Grade **50.00** out of 50.00 (**100%**)

Name [KEERTHI AACHUTHAN K 2022-CSD-A](#)

Question 1

Correct

Mark 1.00 out of 1.00

Write a program to find the eligibility of admission for a professional course based on the following criteria:

Marks in Maths ≥ 65

Marks in Physics ≥ 55

Marks in Chemistry ≥ 50

Or

Total in all three subjects ≥ 180

Sample Test Cases

Test Case 1

Input

70

60

80

Output

The candidate is eligible

Test Case 2

Input

50

80

80

Output

The candidate is eligible

Test Case 3

Input

50

60

40

Output

The candidate is not eligible

For example:

Input	Result
70	The candidate is eligible
60	
80	

Answer: (penalty regime: 0 %)

```

1 maths = int(input())
2 physics = int(input())
3 chemistry = int(input())
4 total = maths + physics + chemistry
5 if maths >= 65 and physics >= 55 and chemistry >= 50:
6     print("The candidate is eligible")
7 elif total >= 180:
8     print("The candidate is eligible")
9 else:
10    print("The candidate is not eligible")
11

```

	Input	Expected	Got	
✓	70 60 80	The candidate is eligible	The candidate is eligible	✓
✓	50 80 80	The candidate is eligible	The candidate is eligible	✓
✓	50 60 40	The candidate is not eligible	The candidate is not eligible	✓
✓	20 10 25	The candidate is not eligible	The candidate is not eligible	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question 2

Correct

Mark 1.00 out of 1.00

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

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

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

Sample Input 1

1900

Sample Output 1

1900 is not a leap year.

Sample Input 2

2000

Sample Output 2

2000 is a leap year.

Answer: (penalty regime: 0 %)

```

1 year = int(input())
2 if year % 400 == 0:
3     print(year, "is a leap year.")
4 elif year % 100 == 0:
5     print(year, "is not a leap year.")
6 elif year % 4 == 0:
7     print(year, "is a leap year.")
8 else:
9     print(year, "is not a leap year.")
10

```

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

Passed all tests! ✓

Correct

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 %)

```
1 num1 = int(input())
2 operator = input()
3 num2 = int(input())
4
5 if operator == '+':
6     result = num1 + num2
7 elif operator == '-':
8     result = num1 - num2
9 elif operator == '/':
10    result = num1 / num2
11 elif operator == '*':
12    result = num1 * num2
13 print(result)
14
```

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

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question 4

Correct

Mark 1.00 out of 1.00

In 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	False
43	

Answer: (penalty regime: 0 %)

```

1 | weapons = int(input())
2 | soldiers = int(input())
3 v if weapons % 3 == 0 and soldiers % 2 == 0:
4 |   print(True)
5 v else:
6 |   print(False)
7 |

```

	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 5

Correct

Mark 1.00 out of 1.00

A triangle can be classified based on the lengths of its sides as equilateral, isosceles or scalene. All three sides of an equilateral triangle have the same length. An isosceles triangle has two sides that are the same length, and a third side that is a different length. If all of the sides have different lengths then the triangle is scalene.

Write a program that reads the lengths of the three sides of a triangle from the user. Then display a message that states the triangle's type.

Sample Input 1

```
60
60
60
```

Sample Output 1

That's a equilateral triangle

Sample Input 2

```
40
40
80
```

Sample Output 2

That's a isosceles triangle

Sample Input 3

```
50
60
70
```

Sample Output 3

That's a scalene triangle

For example:

Input	Result
60 60 60	That's a equilateral triangle
40 40 80	That's a isosceles triangle

Answer: (penalty regime: 0 %)

```

1 side1 = int(input())
2 side2 = int(input())
3 side3 = int(input())
4
5 if side1 == side2 == side3:
6     print("That's a equilateral triangle")
7 elif side1 == side2 or side1 == side3 or side2 == side3:
8     print("That's a isosceles triangle")
9 else:
10    print("That's a scalene triangle")
11
```

	Input	Expected	Got	
✓	60 60 60	That's a equilateral triangle	That's a equilateral triangle	✓
✓	40 40 80	That's a isosceles triangle	That's a isosceles triangle	✓
✓	50 60 70	That's a scalene triangle	That's a scalene triangle	✓
✓	50 50 80	That's a isosceles triangle	That's a isosceles triangle	✓
✓	10 10 10	That's a equilateral triangle	That's a equilateral triangle	✓

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 Wednesday, 13 March 2024, 10:30 AM

State Finished

Completed on Friday, 29 March 2024, 8:31 PM

Time taken 16 days 10 hours

Marks 5.00/5.00

Grade **50.00** out of 50.00 (**100%**)

Name [KEERTHI AACHUTHAN K 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 %)

```
1 # Initialize variables
2 total = 0
3 count = 0
4
5 # Take input from the user
6 num = float(input())
7
8 # Check if the first input is 0
9 if num == 0:
10     print("Error: First value cannot be 0.")
11 else:
12     # Continue taking input until 0 is entered
13     while num != 0:
14         total += num
15         count += 1
16         num = float(input())
17
18 # Calculate the average
19 average = total / count
20
21 # Display the average with format specifier
22 print("The average is {:.1f}.".format(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 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 x_1 and moves at a speed v_1 meters per jump.
- The second kangaroo starts at position x_2 and moves at a speed of v_2 meters per jump and $x_2 > x_1$
- 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:

x_1 -position of kangaroo1
 v_1 -Speed of kangaroo1
 x_2 -position of kangaroo2
 v_2 -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	YES
3	
4	
2	
6	

Answer: (penalty regime: 0 %)

```

1 x1 = int(input())
2 v1 = int(input())
3 x2 = int(input())
4 v2 = int(input())
5 k = int(input())
6
7 for i in range(0,k):
8     x1 += v1
9     x2 += v2
10    if(x1 == x2):
11        break;
12    if(i<k-1):
13        print("YES")
14    else:
15        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 sum of the series $1 + 11 + 111 + 1111 + \dots + n$ terms (n will be given as input from the user and sum will be the output)

Sample Test Cases

Test Case 1

Input

4

Output

1234

Explanation:

as input is 4, have to take 4 terms.

1 + 11 + 111 + 1111

Test Case 2

Input

6

Output

123456

For example:

Input	Result
3	123

Answer: (penalty regime: 0 %)

```

1 |def recSum(n):
2 |    if n<=0:
3 |        return 0
4 |    else:
5 |        return (n + 10 * recSum(n-1))
6 |
7 |# Reading number of terms
8 |term = int(input(""))
9 |series_sum = recSum(term)
10|print(series_sum)

```

	Input	Expected	Got	
✓	1	1	1	✓
✓	3	123	123	✓

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

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question 4

Correct

Mark 1.00 out of 1.00

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

Sample Input

10

Sample Output

The sum of the first 10 positive integers is 55.0

For example:

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

Answer: (penalty regime: 0 %)

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

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

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question 5

Correct

Mark 1.00 out of 1.00

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 %)

```

1 n = int(input())
2 count = 0
3 while n > 0:
4     n = n//10
5     count += 1
6 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.

[Week-04_MCQ](#)

Jump to...

[WEEK-04-Extra ►](#)

[Dashboard](#) / My courses / [CD19411-PPD-2022](#) / [WEEK_05-Lists](#) / [WEEK-05_CODING](#)

Started on Wednesday, 24 April 2024, 8:55 PM

State Finished

Completed on Thursday, 25 April 2024, 10:48 PM

Time taken 1 day 1 hour

Marks 5.00/5.00

Grade **50.00** out of 50.00 (**100%**)

Name [KEERTHI AACHUTHAN K 2022-CSD-A](#)

Question 1

Correct

Mark 1.00 out of 1.00

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

Input :

1 2 1 3 4 3

3

Output :

2

3

3

2

Explanation

All windows of size K are

[1, 2, 1]

[2, 1, 3]

[1, 3, 4]

[3, 4, 3]

Answer: (penalty regime: 0 %)

```

1 arr = [1, 2, 1, 3, 4, 3]
2 k = 3
3
4 for i in range(len(arr) - k + 1):
5     window = arr[i:i+k]
6     distinct_count = len(set(window))
7     print(distinct_count)
8

```

	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

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	10
30	20
40	30
50	40
10	50
0	

Answer: (penalty regime: 0 %)

```
1 n = 1  
2 l = []  
3 while n != 0:  
4     n = int(input())  
5     if(n == 0):  
6         break  
7     l.append(n)  
8 l.sort()  
9 for i in l:  
10    print(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 3

Correct

Mark 1.00 out of 1.00

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

Sample Input

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

Sample Output

```
-15
-20
0
5
10
```

For example:

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

Answer: (penalty regime: 0 %)

```
1 a=[]
2 while True:
3     n=int(input())
4     if(n==(-99)):
5         break
6     a.append(n)
7 for i in a:
8     if(i<0):
9         print(i)
10 for i in a:
11     if(i==0):
12         print(i)
13 for i in a:
14     if(i>0):
15         print(i)
```

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

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

//

Question 4

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	True
12 25 85 40 21	

Answer: (penalty regime: 0 %)

```

1 a=input()
2 b=input()
3 ls1=list(a.split(" "))
4 ls2=list(b.split(" "))
5 count=0
6 for i in ls1:
7     for j in ls2:
8         if i==j:
9             count=count+1
10 if(count!=0):
11     print("True")
12 else:
13     print("False")
14

```

	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 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	

Answer: (penalty regime: 0 %)

```

1 import math
2 n=int(input())
3 a=[]
4 for i in range (0,n):
5     b=int(input())
6     a.append(b)
7 sum=0
8 for i in range(0,n):
9     for i in range (0,i+1):

```

```
10 |     temp=a[i]%10
11 |     a[i]=a[i]/10
12 |     temp=math.floor(temp)
13 |     sum=sum+temp*temp
14 | print(sum)
```

	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-05_MCQ

Jump to...

WEEK-05-Extra ►

Started on Saturday, 18 May 2024, 10:40 AM

State Finished

Completed on Saturday, 18 May 2024, 10:53 AM

Time taken 13 mins 17 secs

Marks 5.00/5.00

Grade 50.00 out of 50.00 (100%)

Name [KEERTHI AACHUTHAN K 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 %)

```

1 def last(string):
2     word = string.split()
3     lastw= word[-1]
4     last= len(lastw)
5     return last
6
7 str=input()
8 print(last(str))

```

	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 %)

```

1 a=str(input())
2 b=['break','case','continue','default','defer','else','for','func','goto','if','map',
3 if a in b:
4     print("%s is a keyword"%(a))
5 else:
6     print("%s is not a keyword"%(a))
7

```

	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 %)

```

1 a=str(input())
2 b=['break','case','continue','default','defer','else','for','func','goto','if','map',
3 if a in b:
4     print("%s is a keyword"%(a))
5 else:
6     print("%s is not a keyword"%(a))
7

```

	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

Verify the given number is cyclic or not.

Input Format

Num1

Num2

Constraints

1<=range<=9999999999

Sample Input 1

12345

45123

Sample Output 1

Yes

Sample Input 2

12345

54123

Sample Output 2

No

Answer: (penalty regime: 0 %)

```

1 def is_cyclic(num1, num2):
2     str_num1 = str(num1)
3     str_num2 = str(num2)
4
5
6     if len(str_num1) != len(str_num2):
7         return "No"
8
9
10    doubled_num1 = str_num1 + str_num1
11
12
13    if doubled_num1.find(str_num2) != -1:
14        return "Yes"
15    else:
16        return "No"
17
18    num1 = input()
19    num2 = input()
20
21 result = is_cyclic(num1, num2)

```

	Input	Expected	Got	
✓	12345 45123	Yes	Yes	✓
✓	12345 54123	No	No	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question 5

Correct

Mark 1.00 out of 1.00

In a particular jurisdiction, older license plates consist of three uppercase letters followed by three digits. When all of the license plates following that pattern had been used, the format was changed to four digits followed by three uppercase letters.

Write a program that begins by reading a string of characters from the user. Then your program should display a message indicating whether the characters are valid for an older style license plate or a newer style license plate. Your program should display an appropriate message if the string entered by the user is not valid for either style of license plate.

Sample Input 1

ABC123

Sample Output 1

The plate is a valid older style plate.

Sample Input 2

123ABCD

Sample Output 2

The plate is not valid.

Sample Input 2

1234ABC

Sample Output 3

The plate is a valid newer style plate.

Answer: (penalty regime: 0 %)

```

1 def check_license_plate(plate):
2     if len(plate) == 6 and plate[:3].isalpha() and plate[3:].isdigit():
3         return "The plate is a valid older style plate."
4     elif len(plate) == 7 and plate[:4].isdigit() and plate[4:7].isalpha():
5         return "The plate is a valid newer style plate."
6     else:
7         return "The plate is not valid."
8
9 plate = input()
10
11 print(check_license_plate(plate))
12

```

	Input	Expected	Got	
✓	ABC123	The plate is a valid older style plate.	The plate is a valid older style plate.	✓
✓	123ABCD	The plate is not valid.	The plate is not valid.	✓
✓	1234ABC	The plate is a valid newer style plate.	The plate is a valid newer style plate.	✓

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, 8:18 PM

State Finished

Completed on Saturday, 18 May 2024, 11:13 AM

Time taken 8 days 14 hours

Marks 5.00/5.00

Grade **50.00** out of 50.00 (**100%**)

Name [KEERTHI AACHUTHAN K 2022-CSD-A](#)

Question 1

Correct

Mark 1.00 out of 1.00

A list rotation consists of taking the last element and moving it to the front. For instance, if we rotate the list [1,2,3,4,5], we get [5,1,2,3,4]. If we rotate it again, we get [4,5,1,2,3].

Write a Python function `rotatelist(l,k)` that takes a list `l` and a positive integer `k` and returns the list `l` after `k` rotations. If `k` is not positive, your function should return `l` unchanged. Note that your function should not change `l` itself, and should return the rotated list.

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

```
>>> rotatelist([1,2,3,4,5],1)
[5, 1, 2, 3, 4]
```

```
>>> rotatelist([1,2,3,4,5],3)
[3, 4, 5, 1, 2]
```

```
>>> rotatelist([1,2,3,4,5],12)
[4, 5, 1, 2, 3]
```

Answer: (penalty regime: 0 %)

[Reset answer](#)

```

1 def Reverse(l, start , end):
2     while start < end:
3         temp = l[start]
4         l[start] = l[end]
5         l[end] = temp
6         start += 1
7         end -= 1
8
9 def rotatelist(l,k):
10    k = k % len(l)
11    Reverse(l , 0 , len(l) - 1)
12    Reverse(l , 0 , k - 1)
13    Reverse(l , k, len(l) - 1)
14    return l
15
```

	Test	Expected	Got	
✓	<code>print(rotatelist([1,2,3,4,5],1))</code>	[5, 1, 2, 3, 4]	[5, 1, 2, 3, 4]	✓
✓	<code>print(rotatelist([1,2,3,4,5],3))</code>	[3, 4, 5, 1, 2]	[3, 4, 5, 1, 2]	✓
✓	<code>print(rotatelist([1,2,3,4,5],12))</code>	[4, 5, 1, 2, 3]	[4, 5, 1, 2, 3]	✓

Passed all tests! ✓

[Correct](#)

Marks for this submission: 1.00/1.00.

Question 2

Correct

Mark 1.00 out of 1.00

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

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

Sample Input

```
5  
10  
15  
20  
25
```

Sample Output

```
75
```

Answer: (penalty regime: 0 %)[Reset answer](#)

```
1 def read_and_sum():  
2     value = input().strip()  
3     if value == '':  
4         return 0  
5     return int(value) + read_and_sum()  
6 print(read_and_sum())  
7
```

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

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question 3

Correct

Mark 1.00 out of 1.00

Given an integer n , return a list of length $n + 1$ such that for each i ($0 \leq i \leq n$), $\text{ans}[i]$ is the number of 1's in the binary representation of i .

Example:

Input: $n = 2$
Output: $[0, 1, 1]$
Explanation:
 $0 \rightarrow 0$
 $1 \rightarrow 1$
 $2 \rightarrow 10$

Example2:

Input: $n = 5$
Output: $[0, 1, 1, 2, 1, 2]$
Explanation:
 $0 \rightarrow 0$
 $1 \rightarrow 1$
 $2 \rightarrow 10$
 $3 \rightarrow 11$
 $4 \rightarrow 100$
 $5 \rightarrow 101$

Note: Complete the given function alone

For example:

Test	Result
<code>print(CountingBits(5))</code>	<code>[0, 1, 1, 2, 1, 2]</code>

Answer: (penalty regime: 0 %)[Reset answer](#)

```

1 def CountingBits(n):
2     result = []
3     for i in range(n + 1):
4         count = 0
5         while i:
6             count += i & 1
7             i >>= 1
8         result.append(count)
9     return result
10
11

```

	Test	Expected	Got	
✓	<code>print(CountingBits(2))</code>	<code>[0, 1, 1]</code>	<code>[0, 1, 1]</code>	✓
✓	<code>print(CountingBits(5))</code>	<code>[0, 1, 1, 2, 1, 2]</code>	<code>[0, 1, 1, 2, 1, 2]</code>	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question 4

Correct

Mark 1.00 out of 1.00

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

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

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

```
True
```

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

```
False
```

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

```
True
```

Answer: (penalty regime: 0 %)

```

1 from math import *
2
3 def issquare(n):
4     k = int(sqrt(n))
5     return(k*k == n)
6
7 def sumofsquares(m):
8     for num in range(m//2):
9         res=issquare(num) and issquare(m-num)
10        if res==True:
11            break
12    return res

```

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

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question 5

Correct

Mark 1.00 out of 1.00

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

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

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

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

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

```
False
```

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

```
True
```

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

```
False
```

Answer: (penalty regime: 0 %)

[Reset answer](#)

```
1 def wellbracketed(s):
2     depth = 0
3     for char in s:
4         if char == '(':
5             depth += 1
6         elif char == ')':
7             depth -= 1
8         if depth < 0:
9             return False
10    return depth == 0
```

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

Passed all tests! ✓

[Correct](#)

Marks for this submission: 1.00/1.00.

[◀ Week-07_MCQ](#)

[Jump to...](#)[WEEK-07-Extra ►](#)

[Dashboard](#) / My courses / [CD19411-PPD-2022](#) / [WEEK_08-Tuple](#) / [WEEK-08_CODING](#)

Started on Monday, 13 May 2024, 6:48 PM

State Finished

Completed on Saturday, 18 May 2024, 11:31 AM

Time taken 4 days 16 hours

Marks 5.00/5.00

Grade **50.00** out of 50.00 (**100%**)

Name [KEERTHI AACHUTHAN K 2022-CSD-A](#)

Question 1

Correct

Mark 1.00 out of 1.00

A customer wants to buy a mobile phone in a online mart, the customer finds different prices from different seller, the item price is been stored in a nested tuples in the following order ((seller_name,item_name,item_cost)), consider the tuple has 5 seller, write a program to help the customer to view in the order of lowest price of item first and so on.

sample input:

seller_1

samsung

45000.00

seller_2

samsung

45500.00

seller_3

samsung

44700.00

seller_4

samsung

43900.00

seller_5

samsung

44100.00

sample output:

```
(("seller_4","samsung","43900.00"),("seller_5","samsung","44100.00"),("seller_3","samsung","44700.00"),
("seller_1","samsung","45000.00"),("seller_2","samsung","45500.00"))
```

Answer: (penalty regime: 0 %)

```
1 sellers=[]
2 for _ in range(5):
3     seller_name=input()
4     item_name=input()
5     item_cost=float(input())
6     sellers.append((seller_name,item_name,"{:.2f}".format(item_cost)))
7 sorted_sellers=sorted(sellers,key=lambda x: x[2])
8 print(tuple(sorted_sellers))
```

	Input	Expected	Got	
✓	seller_1 samsung 45000.00 seller_2 samsung 45500.00 seller_3 samsung 44700.00 seller_4 samsung 43900.00 seller_5 samsung 44100.00	('seller_4', 'samsung', '43900.00'), ('seller_5', 'samsung', '44100.00'), ('seller_3', 'samsung', '44700.00'), ('seller_1', 'samsung', '45000.00'), ('seller_2', 'samsung', '45500.00'))	('seller_4', 'samsung', '43900.00'), ('seller_5', 'samsung', '44100.00'), ('seller_3', 'samsung', '44700.00'), ('seller_1', 'samsung', '45000.00'), ('seller_2', 'samsung', '45500.00'))	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question 2

Correct

Mark 1.00 out of 1.00

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

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

Answer: (penalty regime: 0 %)

```
1 empty_tuple = ()  
2 print( empty_tuple)  
3  
4 int_tuple = (4, 5, 6)  
5 print( int_tuple)  
6  
7 mixed_tuple = (1, 'ECE', 'MCT', 'R&A', 3.4)  
8 print( mixed_tuple)  
9  
10 nested_tuple = ('Python', [8, 4, 6], (1, 2, 3))  
11 print( nested_tuple)  
12
```

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

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question 3

Correct

Mark 1.00 out of 1.00

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

sample input:

bread

45

5

milk

40

2

cheese

60

2

butter

90

2

jam

60

2

sample output: 725

Answer: (penalty regime: 0 %)

```
1 items = []
2 for _ in range(5):
3     item_name = input()
4     item_cost = int(input())
5     no_of_item = int(input())
6     items.append((item_name, item_cost, no_of_item))
7
8 total_cost = sum(cost * quantity for _, cost, quantity in items)
9 print(total_cost)
10
```

	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 4

Correct

Mark 1.00 out of 1.00

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

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

Sample Input

Apple

p

Sample Output

2

1

Answer: (penalty regime: 0 %)

```
1 a = input()
2 b = input()
3 print(a.count(b))
4 print(a.find(b))
```

	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 5

Correct

Mark 1.00 out of 1.00

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

Sample Input

Rakalakshmi

a

Sample Output

True

Sample Input

Rakalakshmi

b

Sample Output

False

Answer: (penalty regime: 0 %)

```
1 string = input()
2 char = input()
3 print(char in string)
4
```

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

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 Wednesday, 15 May 2024, 10:42 PM

State Finished

Completed on Saturday, 18 May 2024, 11:43 AM

Time taken 2 days 13 hours

Marks 5.00/5.00

Grade **50.00** out of 50.00 (**100%**)

Name [KEERTHI AACHUTHAN K 2022-CSD-A](#)

Question 1

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 %)

```

1 | a = set(input())
2 | b = set(input())
3 | c = set(input())
4 v if c.issubset(a):
5 |   print("yes\nset3 is subset of set1 and set2")
6 v else:
7 |   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 2

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 %)

```

1 num = input()
2 digit_freq = {}
3 for digit in num:
4     if digit.isdigit():
5         digit_freq[digit] = digit_freq.get(digit, 0) + 1
6 if len(set(digit_freq.values())) == 1:
7     print("Stable Number")
8 else:
9     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.

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 arrayszie

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

Ouput : Phone numbers printed in ascending order

Sample Test Case

Input

5

6

9840403212 9890909012 98123455 90123456 99123456

90909090 99999999 9840403212 12345678 12347890 99123456

Output

12345678 12347890 90123456 90909090 98123455 99123456 99999999 9840403212 9890909012

Answer: (penalty regime: 0 %)

```

1 n=int(input())
2 m=int(input())
3 List1=list(map(int,input().split(" ")))
4 List2=list(map(int,input().split(" ")))
5 for i in List2:
6     List1.append(i)
7 List1=list(set(List1))
8 List1.sort()
9 for i in List1:
10    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! ✓

Correct

Marks for this submission: 1.00/1.00.

Question 4

Correct

Mark 1.00 out of 1.00

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

Input

we are good are we good

Output

are good we

Count = 1

For example:

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

Answer: (penalty regime: 0 %)

```

1 sent = input()
2 sent = sent.split()
3 sent_set = set(sent)
4 unique_words = list(sent_set)
5 unique_words.sort()
6 count = 0
7 for word in unique_words:
8     print(word, end=" ")
9     if len(word) > 3:
10         count += 1
11 print("\nCount =", 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 5

Correct

Mark 1.00 out of 1.00

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

Input :

1 2 1 3 4 3

3

Output :

2

3

3

2

Explanation

All windows of size K are

[1, 2, 1]

[2, 1, 3]

[1, 3, 4]

[3, 4, 3]

Answer: (penalty regime: 0 %)

```

1 def count_distinct_numbers(arr, k):
2     n = len(arr)
3     for i in range(n - k + 1):
4         window = arr[i:i + k]
5         distinct_count = len(set(window))
6         print(distinct_count)
7 arr = list(map(int, input().split()))
8 k = int(input())
9 count_distinct_numbers(arr, k)

```

	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.

[**◀ Week-09_MCQ**](#)[Jump to...](#)[**WEEK-09-Extra ►**](#)

[Dashboard](#) / My courses / [CD19411-PPD-2022](#) / [WEEK 10-Dictionary](#) / [WEEK-10_CODING](#)

Started on Wednesday, 15 May 2024, 10:41 PM

State Finished

Completed on Saturday, 18 May 2024, 11:43 AM

Time taken 2 days 13 hours

Marks 7.00/7.00

Grade 50.00 out of 50.00 (100%)

Name [KEERTHI AACHUTHAN 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 %)

```

1 w1 = input().lower()
2 w2 = input().lower()
3 word1 = ''.join(sorted(w1))
4 word2 = ''.join(sorted(w2))
5 if word1 == word2:
6     print("Those strings are anagrams.")
7 else:
8     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

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

Points Letters

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

2 D and G

3 B, C, M and P

4 F, H, V, W and Y

5 K

8 J and X

10 Q and Z

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

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

Sample Input

REC

Sample Output

REC is worth 5 points.

Answer: (penalty regime: 0 %)

```

1 x = {
2     'A': 1, 'E': 1, 'I': 1, 'L': 1, 'N': 1, 'O': 1, 'R': 1, 'S': 1, 'T': 1, 'U': 1,
3     'D': 2, 'G': 2,
4     'B': 3, 'C': 3, 'M': 3, 'P': 3,
5     'F': 4, 'H': 4, 'V': 4, 'W': 4, 'Y': 4,
6     'K': 5,
7     'J': 8, 'X': 8,
8     'Q': 10, 'Z': 10}
9 word = input().upper()
10 score = sum(x.get(letter, 0) for letter in word)
11 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.

Question 3

Correct

Mark 1.00 out of 1.00

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

For example:

Input Result

Hello, World!	10
---------------	----

Answer: (penalty regime: 0 %)

```
1 x = input()
2 y = set(x)
3 c = x.count(' ')
4 special = sum(1 for char in x if not char.isalnum() and not char.isspace())
5 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 4

Correct

Mark 1.00 out of 1.00

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

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

Enter Key to check: A

Output:

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

Present # True Statement

Not Present # False Statement

Answer: (penalty regime: 0 %)

```
1 |d = {'A':1, 'B':2, 'C':3}
2 |d = input()
3 vif a in d:
4 |    print("Present")
5 velse:
6 |    print("Not Present")
```

	Input	Expected	Got	
✓	A	Present	Present	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question 5

Correct

Mark 1.00 out of 1.00

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:

$2 \leq s.length \leq 200$

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

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

The words in s are separated by a single space.

s contains no leading or trailing spaces.

Answer: (penalty regime: 0 %)

```
1 | s = input().split()
2 vdef pos(word):
3 |     return int(word[-1])
4 | s.sort(key=pos)
5 o = ' '.join(word[:-1] for word in s)
6 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 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 %)

```

1 a = {'a': 10, 'b': 10, 'c': 239}
2 result = 1
3 for value in a.values():
4     result *= value
5 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

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

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

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

Input Format:

In line 1 no of students will be given

Followed by n lines containing student rollno and marks

Output Format:

Line 1 Class average

Line 2 Maximum mark Roll no

Line 3 Minimum mark Roll no

Sample Input:

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

Output:

```
77
02
03
```

Answer: (penalty regime: 0 %)

```
1 n = int(input())
2 total_marks = 0
3 max_marks = -1
4 min_marks = float('inf')
5 max_roll_no = ""
6 min_roll_no = ""
7
8 for i in range(n):
9     roll_no, marks = input().split()
10    marks = int(marks)
11    total_marks += marks
12
13    if marks > max_marks:
14        max_marks = marks
15        max_roll_no = roll_no
16
17    if marks < min_marks:
18        min_marks = marks
19        min_roll_no = roll_no
20
21 class_avg = total_marks // n
22 print(class_avg)
```

	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 ►