

GE19211 / GE23233 / GE23231 - PSPP/PUP

Dashboard / My courses / PSPP/PUP / Experiments based on Strings and its operations. / Week5_Coding

Quiz navigation

1	2	3	4	5	6
✓	✓	✓	✓	✓	✓
7	8	9	10		
✓	✓	✓	✓		

Show one page at a time

Finish review

Started on	Tuesday, 7 May 2024, 1:36 PM
State	Finished
Completed on	Tuesday, 7 May 2024, 9:08 PM
Time taken	7 hours 32 mins
Marks	10.00/10.00
Grade	100.00 out of 100.00

Question 1

Correct

Mark 1.00 out of 1.00

Y Flag question

Write a program that takes as input a string (sentence), and returns its second word in uppercase.

For example:

If input is "Wipro Technologies Bangalore" the function should return "TECHNOLOGIES"

If input is "Hello World" the function should return "WORLD"

If input is "Hello" the program should return "LESS"

NOTE 1: If input is a sentence with less than 2 words, the program should return the word "LESS".

NOTE 2: The result should have no leading or trailing spaces.

For example:

Input	Result
Wipro Technologies Bangalore	TECHNOLOGIES
Hello World	WORLD
Hello	LESS

Answer: (penalty regime: 0 %)

```
1 def get_second_word_in_uppercase(sentence):
2     words = sentence.split()
3     if len(words)>=2:
4         return words[1].upper().strip()
5     else:
6         return "LESS"
7     input_sentence=input().strip()
8     output_word=get_second_word_in_uppercase(input_sentence)
9     print(output_word)
```

Input	Expected	Got	
✓ Wipro Technologies Bangalore	TECHNOLOGIES	TECHNOLOGIES	✓
✓ Hello World	WORLD	WORLD	✓
✓ Hello	LESS	LESS	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question 2

Correct

Mark 1.00 out of 1.00

Y Flag question

In this exercise, you will create a program that reads words from the user until the user enters a blank line. After the user enters a blank line your program should display each word entered by the user exactly once. The words should be displayed in the same order that they were first entered. For example, if the user enters:

first

second

first

third

second

then your program should display:

first

second

third

Answer: (penalty regime: 0 %)

```
1 words = []
2 while True:
3     try:
4         word = input().strip()
5         if not word :
6             break
7         if word not in words:
8             words.append(word)
9     except EOFError:
10        break
11 print(' '.join(words))
```

Input	Expected	Got	
✓ first	first	first	✓
✓ second	second	second	✓
✓ first	first	first	✓
✓ third	third	third	✓
✓ second	second	second	✓
✓ rec	rec	rec	✓
✓ cse	cse	cse	✓
✓ it	it	it	✓
✓ rec	rec	rec	✓
✓ cse	cse	cse	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question 3

Correct

Mark 1.00 out of 1.00

Y Flag question

Given two Strings s1 and s2, remove all the characters from s1 which is present in s2.

Constraints

1<= string length <= 200

Sample Input 1

experience

enc

Sample Output 1

xpri

Answer: (penalty regime: 0 %)

```
1 s1 = input()
2 s2 = input()
3 s2_set = set(s2)
4 result = ''
5 for char in s1:
6     if char not in s2_set:
7         result += char
8     else:
9         continue
10 print(result)
```

Input	Expected	Got	
✓ experience	xpri	xpri	✓
✓ enc			✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question 4

Correct

Mark 1.00 out of 1.00

Y Flag question

Write a python program to count all letters, digits, and special symbols respectively from a given string

For example:

Input	Result
rec@123 3	3
1	1

Answer: (penalty regime: 0 %)

```
1 s = input()
2 l_count=0
3 d_count=0
4 symbols_count=0
5 for char in s:
6     if char.isalpha():
7         l_count+=1
8     elif char.isdigit():
9         d_count+=1
10    elif not char.isalnum():
11        symbols_count+=1
12 print(l_count)
13 print(d_count)
14 print(symbols_count)
```

Input	Expected	Got	
✓ rec@123	3	3	✓
	3	3	✓
	1	1	✓
✓ P@yn2at*815ve	8	8	✓
	3	3	✓
	4	4	✓
✓ abc@128	3	3	✓
	2	2	✓
	2	2	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question 5

Correct

Mark 1.00 out of 1.00

Y Flag question

Write a program to check if two strings are balanced. For example, strings s1 and s2 are balanced if all the characters in the s1 are present in s2. The character's position doesn't matter. If balanced display as "true", otherwise "false".

For example:

Input	Result
Yn	True
Pn@tive	False

Answer: (penalty regime: 0 %)

```
1 def are_strings_balanced(s1,s2):
2     set_s1 = set(s1)
3     set_s2 = set(s2)
4     return set_s1.issubset(set_s2)
5 s1=input()
6 s2=input()
7
8 if are_strings_balanced(s1,s2):
9     print("True")
10 else:
11     print("False")
```

Input	Expected	Got	
✓ Yn	True	True	✓
✓ Pn@tive	False	False	✓
✓ Ynf	False	False	✓
✓ Pn@tive	False	False	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question 6

Correct

Mark 1.00 out of 1.00

Y Flag question

Reverse a string without affecting special characters

Given a string S, containing special characters and all the alphabets, reverse the string without affecting the positions of the special characters.

Input:

A\$B

Output:

B&A

Explanation: As we ignore '&' and

As we ignore '&' and then reverse, so answer is "B&A".

For example:

Input	Result
A&#	a&#

Answer: (penalty regime: 0 %)

```
1 S = input().strip()
2 s_list = list(S)
3 left = 0
4 right = len(s_list)-1
5 while left < right:
6     if s_list[left].isalpha() and s_list[right].isalpha():
7         s_list[left],s_list[right] = s_list[right],s_list[left]
8         left += 1
9         right -= 1
10    elif not s_list[left].isalpha():
11        left += 1
12    elif not s_list[right].isalpha():
13        right -= 1
14    result = ''.join(s_list)
15 print(result)
```

Input	Expected	Got	
✓ A&B	B&A	B&A	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question 7

Correct

Mark 1.00 out of 1.00

Y Flag question

Assume that the given string has enough memory.

Don't use any extra space (IN-PLACE)

Sample Input 1

a2b4c6

Sample Output 1

aaabbbcccccc

Answer: (penalty regime: 0 %)

```
1 def expand_string(input_str):
2     result = ""
3     i = 0
4     while i < len(input_str):
5         char = input_str[i]
6         i += 1
7         count = 0
8         while i < len(input_str) and input_str[i].isdigit():
9             count = count * 10 + int(input_str[i])
10            i += 1
11        result += char * count
12    return result
13 input_str = input()
14 output_str = expand_string(input_str)
15 print(output_str)
16
```

Input	Expected	Got	
✓ a2b4c6	aaabbbcccccc	aaabbbcccccc	✓
✓ a12b3d4	aaaaaaaaaabbddddd	aaaaaaaaaabbddddd	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question 8

Correct

Mark 1.00 out of 1.00

Y Flag question

Two string values S1, S2 are passed as the input. The program must print first N characters present in S1 which are also present in S2.

Input Format:

The first line contains S1.

The second line contains S2.

The third line contains N.

Output Format:

The first line contains the N characters present in S1 which are also present in S2.

Boundary Conditions:

2 <= N <= 10

2 <= Length of S1, S2 <= 1000

Example Input/Output 1:

Input:

abcdbc

cdefghbb

3

Output:

bcd

Notes:

b occurs twice in common but must be printed only once.

Answer: (penalty regime: 0 %)

```
1 s1 = input().strip()
2 s2 = input().strip()
3 n = int(input())
4 res=""
5 for i in range(n):
6     if s1[i] in s2:
7         res+=s1[i]
8     else:
9         continue
10    final=""
11    for letter in res:
12        if letter not in final:
13            final+=letter
14    print(final)
```

Input	Expected	Got	
✓ abcdbc	bcd	bcd	✓
✓ cdefghbb			✓
✓ 3			✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question 9

Correct

Mark 1.00 out of 1.00

Y Flag question

Given a string S which is of the format USERNAME@DOMAIN.EXTENSION, the program must print the EXTENSION, DOMAIN, USERNAME in the reverse order.

Input Format:

The first line contains S.

Output Format:

The first line contains EXTENSION.

The second line contains DOMAIN.

The third line contains USERNAME.

Boundary Condition:

1 <= Length of S <= 100

Example Input/Output 1:

Input:

abcd@gmail.com

Output:

com

gmail

abcd

For example:

Input	Result
arvijayakumar@rajalakshmi.edu.in	edu.in
	rajalakshmi
	arvijayakumar

Answer: (penalty regime: 0 %)

```
1 s = input()
2 if not 1<=len(s)<=100:
3     exit()
4 parts=s.split('@')
5 if len(parts)>2:
6     exit()
7 username=parts[0]
8 domain=parts[1].split('.')
9 if len(domainpart)>2:
10    exit()
11    domain=domainpart[0]
12 extension='.'.join(domainpart[1:])
13 print(extension)
14 print(domain)
15 print(username)
```

Input	Expected	Got	
✓ abcd@gmail.com	com	com	✓
	gmail	gmail	✓
	abcd	abcd	✓
✓ arvijayakumar@rajalakshmi.edu.in	edu.in	edu.in	✓
	rajalakshmi	rajalakshmi	✓
	arvijayakumar	arvijayakumar	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question 10

Correct

Mark 1.00 out of 1.00

Y Flag question

String should contain only the words are not palindromes.

Sample Input 1

Malayalam is my mother tongue

Sample Output 1

is my mother tongue

Answer: (penalty regime: 0 %)

```
1 def remove_palindromes(sentence):
2     return [word for word in sentence.lower().split() if word != word[::-1]]
3 sentence = input()
4 print(' '.join(remove_palindromes(sentence)))
```

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

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Finish review

Contact site support

You are logged in as DIVYADHARSHINI V 2022-BIOMED-A (LOG OUT)

PSPP/PUP

Data retention summary

→ Week5_MCQ

Jump to...

List →