

Quiz navigation

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

Show one page at a time

Finish review

Started on	Tuesday, 23 April 2024, 2:18 PM
State	Finished
Completed on	Friday, 26 April 2024, 9:13 PM
Time taken	3 days 6 hours
Overdue	1 day 6 hours
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

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:**  
AB&A

**Output:**  
B&A

**Explanation:** As we ignore '&' and  
As we ignore '&' and then reverse, so answer is 'B&A'.

**For example:**

Input	Result
AB&A	B&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)
16
```

Input	Expected	Got
✓ AB&A	B&A	B&A
✓		

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

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 3 1

**Answer:** (penalty regime: 0 %)

```
1 input_str = input()
2
3 letters = sum(1 for char in input_str if char.isalpha())
4 digits = sum(1 for char in input_str if char.isdigit())
5 specials = sum(1 for char in input_str if not char.isalnum())
6
7 print(letters)
8 print(digits)
9 print(specials)
10
```

Input	Expected	Got
✓ rec@123	3 3 3 1	3 3 3 1
✓ P@my26at*4isve	8 3 4 4	8 3 4 4
✓ ab#@12&	3 2 2 2	3 2 2 2

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

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

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 4

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

**Answer:** (penalty regime: 0 %)

```
1 a=input()
2 if not 1<len(a)<=100:
3     exit()
4 parts=a.split('@')
5 if len(parts)!=2:
6     exit()
7 username=parts[0]
8 domainpart=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)
16
```

Input	Expected	Got
✓ abcd@gmail.com	com gmail abcd	com gmail abcd

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

Assume that the given string has enough memory.

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

**Sample Input 1**  
a2b4c6

**Sample Output 1**  
aabbbcccccc

**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	aabbbcccccc	aabbbcccccc
✓ a12b3d4	aaaaaaaaaabbddddd	aaaaaaaaaabbddddd

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

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	

**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")
12
```

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

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

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:  
abcdbde  
cdelfghbb  
3  
Output:  
bcd  
**Note:**  
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
5 common_chars = []
6 for char in s1:
7     if char in s2 and char not in common_chars:
8         common_chars.append(char)
9     if len(common_chars) == N:
10        break
11 print(''.join(common_chars))
12
```

Input	Expected	Got
✓ abcdbde cdelfghbb 3	bcd bcd	bcd bcd

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

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 = ''.join(char for char in s1 if char not in s2_set)
5
6 print(result)
7
```

Input	Expected	Got
✓ experience enc	xpri xpri	xpri xpri

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

String should contain only the words are not palindrome.

**Sample Input 1**  
Malayalam is my mother tongue

**Sample Output 1**  
is my mother tongue

**Answer:** (penalty regime: 0 %)

```
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)))
5
6
```

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.

Question 10

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  
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, sep='\n'))
12
```

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

Passed all tests! ✓

**Correct**  
Marks for this submission: 1.00/1.00.

Finish review

→ Week5\_MCQ

Jump to...

List →

Contact site support

You are logged in as ASWATHY K B 2022-BICMED-A (Log out)

PSPP/PUP

Data retention summary