# LP_Practice_weightOfString

Ramya.V | 10 Feb 2023

Test Name::

LP_Practice_weightOfString

Finish State: Normal | Test Taken on: February 10, 2023 12:04:52 PM IST

R

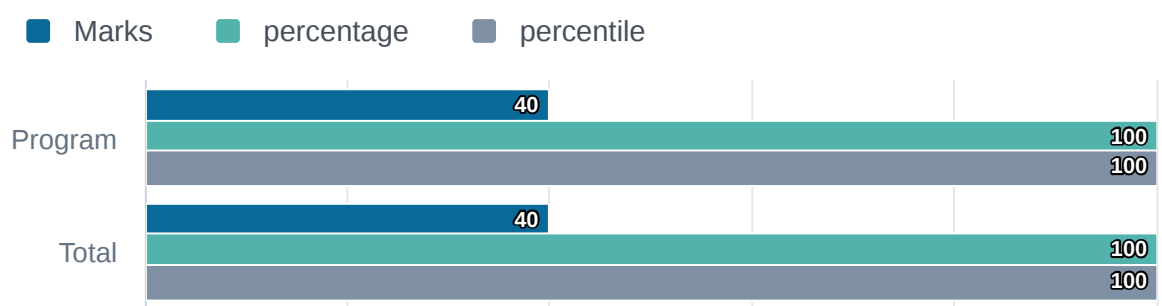
Ramya.V

ramya.v.2020.cse@ritchennai.edu.in

## Overall Summary

| | | |
|---|---|---|
| 40 | Marks Scored | out of 40 |
| 100 % | 100 percentile | out of 47359 Test Takers |
| 6m 48s | Time taken | of 1hr 5mins |

### Marks Scored




### Attempt Summary

Distribution of questions attempted in a total of 1 question(s).

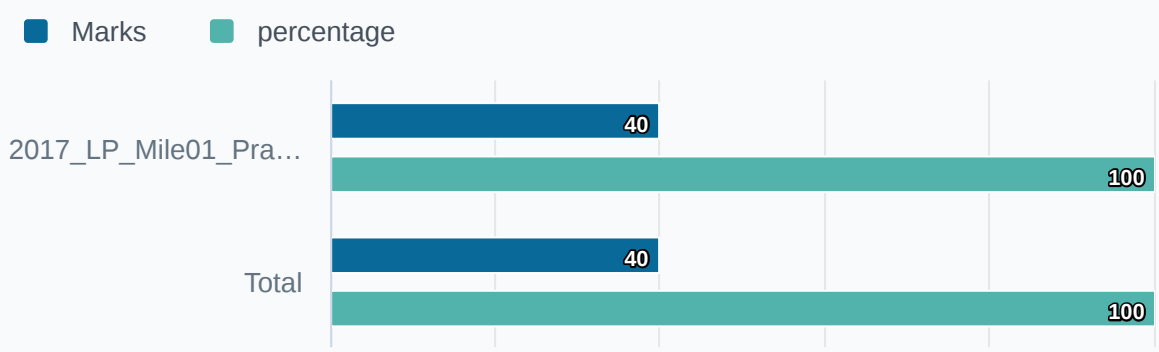

**This shows the correctness of questions attempted by the test taker**

| | | |
|---|---|---|
| Correct | 1 Ques | 40/40 Marks |
| Incorrect | 0 Ques | 0/0 Marks |
| Partially Correct | 0 Ques | 0/0 Marks |
| Not Attempted | 0 Ques | 0/0 Marks |

## Section-Wise Details

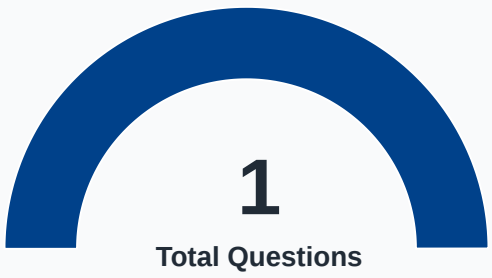
| Section 1 | question(s) | Time taken | Marks Scored |
|---|---|---|---|
| Program | 1 Q. | 6m 48s (Untimed) | 40 / 40 |

### Marks Scored




### Attempt Summary

Distribution of questions attempted in a total of 1 question(s).




This shows the correctness of questions attempted by the test taker

| | | |
|---|---|---|
| Correct | 1 Ques | 40/40 Marks |

## Test Log

10th Feb 2023

| | | |
|---|---|---|
| 11:48 AM |  | Started the test with Program |
| 11:54 AM |  | Finished the test |

## About the Report

This Report is generated electronically on the basis of the inputs received from the assessment takers. This Report including the AI flags that are generated in case of availing of proctoring services, should not be solely used/relied on for making any business, selection, entrance, or employment-related decisions. Mettl accepts no liability from the use of or any action taken or refrained from or for any and all business decisions taken as a result of or reliance upon anything, including, without limitation, information, advice, or AI flags contained in this Report or sources of information used or referred to in this Report.

Mercer | mettl | Ramya.V
LP_Practice_weightOfString / Saved: 0 seconds ago
Test Time: 00:59:04 | Finish Test

1. Program

1

Attempted: 1/1

## Question 1

Revisit Later

How to Attempt?

**Weight of String:** Write a function that takes a string as input and calculates the weight of the string as per rules mentioned below.

For calculating the weight of the string,

- Weight of all alphabetic characters that appear in the string should be added
- Weight of vowels that appear in the string should either be ignored OR added depending upon a specified option
- All non-alphabetic characters in the string should be ignored
- Weight of each letter is its position in the English alphabet system, i.e. weight of a=1, weight of b=2, weight of c=3, weight of d=4, and so on....weight of y=25, weight of z=26.
- Weight of Upper-Case and Lower-Case letters should be taken as the same, i.e. weight of A=a=1, weight of B=b=2, weight of C=c=3, and so on...weight of Z=z=26.

**Example1:**
Let us assume the word is "Hello World!!" and vowels are to be ignored.
Weight of "Hello World!!" = 8+0+12+12+0+0+23+0+18+12+4+0+0 = 89
**Note:** Note that weight of vowels is ignored. Also note that the weight of non-alphabetic characters such as space character and ! is taken as zero.

**Example2:**
Let us assume the word is "Hello World" and vowels are to be included.
Weight of "Hello World!!" = 8+5+12+12+15+0+23+15+18+12+4+0+0 = 124

JAVA7 | Compiler: Java - 1.7

```
1  import java.io.*;
2  import  java.util.*;
3
4  // Read only region start
5  class UserMainCode
6  {
7
8      public int weightOfString(String input1,int input2){
9          // Read only region end
10     {
11 String str=input1.toUpperCase();
12         int sum=0;
13         for(int i=0;i<input1.length();i++)
14         {
15           if(input2==0)
16           {
17       if(str.charAt(i)=='A' || str.charAt(i)=='E' ||str.charAt(i)=='I' || str.charAt(i)=='O'
18                 {
19                     continue;
20                 }
21                 else
22                 {
23                     int a=str.charAt(i)-64;
24                     sum+=a;
25                 }
26           }
```

Use Custom Input | Compile and Test | Submit Code

1. Program

1

Attempted: 1/1

## Question 1

Revisit Later

How to Attempt?

**Weight of String:** Write a function that takes a string as input and calculates the weight of the string as per rules mentioned below.

For calculating the weight of the string,

- Weight of all alphabetic characters that appear in the string should be added
- Weight of vowels that appear in the string should either be ignored OR added depending upon a specified option
- All non-alphabetic characters in the string should be ignored
- Weight of each letter is its position in the English alphabet system, i.e. weight of a=1, weight of b=2, weight of c=3, weight of d=4, and so on....weight of y=25, weight of z=26.
- Weight of Upper-Case and Lower-Case letters should be taken as the same, i.e. weight of A=a=1, weight of B=b=2, weight of C=c=3, and so on...weight of Z=z=26.

**Example1:**
Let us assume the word is "Hello World!!" and vowels are to be ignored.
Weight of "Hello World!!" = 8+0+12+12+0+0+23+0+18+12+4+0+0 = 89
**Note:** Note that weight of vowels is ignored. Also note that the weight of non-alphabetic characters such as space character and ! is taken as zero.

**Example2:**
Let us assume the word is "Hello World" and vowels are to be included.
Weight of "Hello World!!" = 8+5+12+12+15+0+23+15+18+12+4+0+0 = 124

JAVA7 | Compiler: Java - 1.7

```
17      if(str.charAt(i)=='A' || str.charAt(i)=='E' ||str.charAt(i)=='I' || str.charAt(i)=='O'
18              {
19                  continue;
20              }
21              else
22              {
23                  int a=str.charAt(i)-64;
24                  sum+=a;
25              }
26          }
27          else
28          {
29              if(!Character.isLetter(str.charAt(i)))
30                  continue;
31              else
32              {
33                  int a=str.charAt(i)-64;
34                  sum+=a;
35              }
36          }
37      }
38      return sum;
39    }
40  }
41  }
42
```

Use Custom Input

Compile and Test | Submit Code

1. Program

1

Attempted: 1/1

## Question 1

Revisit Later

How to Attempt?

**Weight of String:** Write a function that takes a string as input and calculates the weight of the string as per rules mentioned below.

For calculating the weight of the string,

- Weight of all alphabetic characters that appear in the string should be added
- Weight of vowels that appear in the string should either be ignored OR added depending upon a specified option
- All non-alphabetic characters in the string should be ignored
- Weight of each letter is its position in the English alphabet system, i.e. weight of a=1, weight of b=2, weight of c=3, weight of d=4, and so on....weight of y=25, weight of z=26.
- Weight of Upper-Case and Lower-Case letters should be taken as the same, i.e. weight of A=a=1, weight of B=b=2, weight of C=c=3, and so on...weight of Z=z=26.

**Example1:**
Let us assume the word is "Hello World!!" and vowels are to be ignored.
Weight of "Hello World!!" = 8+0+12+12+0+0+23+0+18+12+4+0+0 = 89
**Note:** Note that weight of vowels is ignored. Also note that the weight of non-alphabetic characters such as space character and ! is taken as zero.

**Example2:**
Let us assume the word is "Hello World" and vowels are to be included.
Weight of "Hello World!!" = 8+5+12+12+15+0+23+15+18+12+4+0+0 = 124

default

CODE EXECUTION DETAILS

Time: 154 ms

Memory: 103812 kb

TEST CASE INFORMATION

Input

Hello World,0

Expected Output

89

Actual Output

89

CONSOLE OUTPUT

STANDARD ERROR/WARNING

None

default2