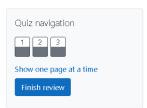
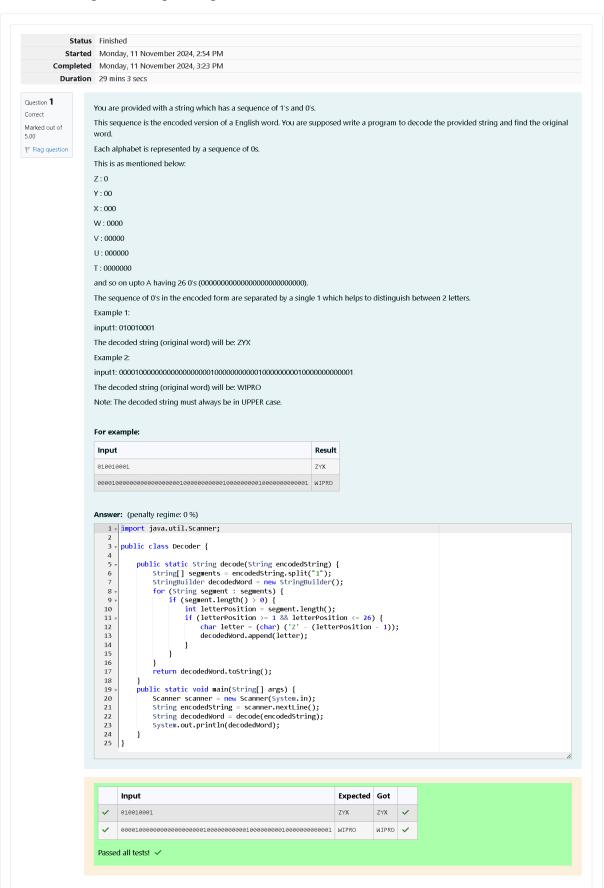
CS23333-Object Oriented Programming Using Java-2023





Question **2** Correct Marked out of 5.00

Given two char arrays input1[] and input2[] containing only lower case alphabets, extracts the alphabets which are present in both arrays (common alphabets).

Get the ASCII values of all the extracted alphabets.

Calculate sum of those ACCII values. Late call it sum 1 and calculate single digit sum of sum 1. i.g., keep adding the digits of sum 1 until you arrive

r riag question

calculate sum of those Aschivations, Lets can it summand calculate single digit sum of summan, key keep adding the digits of summand at a single digit.

Return that single digit as output.

Note:

- 1. Array size ranges from 1 to 10.
- 2. All the array elements are lower case alphabets.
- 3. Atleast one common alphabet will be found in the arrays.

Example 1:

input1: {'a', 'b', 'c'}

input2: {'b', 'c'}

output: 8 Explanation:

'b' and 'c' are present in both the arrays.

ASCII value of 'b' is 98 and 'c' is 99.

98 + 99 = 197 1 + 9 + 7 = 17 1 + 7 = 8

For example:

Input Result

Answer: (penalty regime: 0 %)

```
1 - import java.util.HashSet;
      import java.util.Set;
     public class CommonCharSum {
                cross commontains in {
    public static int calculateSingleDigitSum(char[] input1, char[] input2) {
    Set<Character> set1 = new HashSet<>();
    Set<Character> set2 = new HashSet<>();
                      for (char ch : input1) {
set1.add(ch);
                            for (char ch : input2) {
11
12
                      set2.add(ch);
13
                }
                            set1.retainAll(set2);
14
                int sum1 = 0;
for (char ch : set1) {
15
16
                       sum1 += (int) ch;
17
                }
18
19
                 return getSingleDigitSum(sum1);
20
22
23
           public static int getSingleDigitSum(int sum) {
24
25
                 while (sum >= 10) {
    int tempSum = 0;
26
27
                      while (sum > 0) {
    tempSum += sum % 10;
28
                            sum /= 10;
29
                      sum = tempSum;
31
                 }
32
33
                 return sum;
           public static void main(String[] args){
   char[] input1 = {'a', 'b', 'c'};
   char[] input2 = {'b', 'c'};
35
36
37
38
39
                 int result = calculateSingleDigitSum(input1, input2);
                 System.out.println(result);
40
41
42
```

```
Input Expected Got

| a b c | 8 | 8 | | |
| Passed all tests! | |
```

Question **3**Correct
Marked out of 5.00

Flag question

Write a function that takes an input String (sentence) and generates a new String (modified sentence) by reversing the words in the original String, maintaining the words position.

In addition, the function should be able to control the reversing of the case (upper or lowercase) based on a case_option parameter, as follows:

If case_option = 0, normal reversal of words i.e., if the original sentence is "Wipro TechNologies BangaLore", the new reversed sentence should be "orpiW seigoloNhceT eroLagnaB".

If case_option = 1, reversal of words with retaining position's case i.e., if the original sentence is "Wipro TechNologies BangaLore", the new reversed sentence should be "Orpiw SeigOlonhcet ErolaGnab".

Note that positions 1, 7, 11, 20 and 25 in the original string are uppercase W, T, N, B and L.

Similarly, positions 1, 7, 11, 20 and 25 in the new string are uppercase O, S, O, E and G.

NOLE

- 1. Only space character should be treated as the word separator i.e., "Hello World" should be treated as two separate words, "Hello" and "World". However, "Hello,World", "Hello,World", "Hello,World" or "Hello,World" should be considered as a single word.
- 2. Non-alphabetic characters in the String should not be subjected to case changes. For example, if case option = 1 and the original sentence is "Wipro TechNologies, Bangalore" the new reversed sentence should be "Orpiw,seiGolonhceT Erolagnab". Note that comma has been treated as part of the word "Technologies," and when comma had to take the position of uppercase T it remained as a comma and uppercase T took the position of comma. However, the words "Wipro and Bangalore" have changed to "Orpiw" and "Erolagnab".
- 3. Kindly ensure that no extra (additional) space characters are embedded within the resultant reversed String.

Examples:

S. No.	input1	input2	output
1	Wipro Technologies Bangalore	0	orpiW seigolonhceT erolagnaB
2	Wipro Technologies, Bangalore	0	orpiW ,seigolonhceT erolagnaB
3	Wipro Technologies Bangalore	1	Orpiw Seigolonhcet Erolagnab
4	Wipro Technologies, Bangalore	1	Orpiw ,seigolonhceT Erolagnab

For example:

Input	Result
Wipro Technologies Bangalore 0	orpiW seigolonhceT erolagnaB
Wipro Technologies, Bangalore 0	orpiW ,seigolonhceT erolagnaВ
Wipro Technologies Bangalore 1	Orpiw Seigolonhcet Erolagnab
Wipro Technologies, Bangalore	Огріw ,seigolonhceT Erolagnab

Answer: (penalty regime: 0 %)

```
1 - import java.util.Scanner;
       public class ReverseWordsWithCaseControl {
 3 -
             public static string reverseWordsWithCase(String sentence, int case_option) {
   String[] words = sentence.split(" ");
   StringBuilder result = new StringBuilder();
                      for (String word : words) {
                             if (case_option == 1) {
    reversedWord = reverseWithOriginalPosition(reversedWord, word);
10
12
13
                           result.append(reversedWord).append(" ");
14
                     }return result.toString().trim();
15
16
17
              private static String reverseWord(String word) {
18
19
                     StringBuilder reversed = new StringBuilder(word);
return reversed.reverse().toString();
20
21
              private static String reverseCaseWithOriginalPosition(String reversedWord, String originalWord) {
   StringBuilder result = new StringBuilder(reversedWord);
   for (int i = 0; i < originalWord.length(); i++) {
      char originalLhar = originalWord.charAt(i);
      char reversedChar = reversedWord.charAt(i);
   if (charter interestrectionis); if (charter interestrection);
}</pre>
22
23
25
                            if (Character.isUpperCase(originalChar)) {
    result.setCharAt(i, Character.toUpperCase(reversedChar));
27
28
29
                            } else if (Character.isLowerCase(originalChar)) {
    result.setCharAt(i, Character.toLowerCase(reversedChar));
30
31
32
33
         return result.toString();
34
35
              public static void printResult(String input, int case_option) {
   String result = reverseWordsWithCase(input, case_option);
   System.out.println(result);
36
37
38
40
41
              public static void main(String[] args) {
                     Scanner scanner = new Scanner(System.in);
String input = scanner.nextLine();
42
43
                     int case_option = scanner.nextInt();
printResult(input, case_option);
44
45
46
                      scanner.close();
47
              }
48
```

	Input	Expected	Got	
~	Wipro Technologies Bangalore 0	orpiW seigolonhceT erolagnaB	orpiW seigolonhceT erolagnaB	~
~	Wipro Technologies, Bangalore 0	orpiW ,seigolonhceT erolagnaB	orpiW ,seigolonhceT erolagnaB	~
~	Wipro Technologies Bangalore	Огріы Seigolonhcet Erolagnab	Огріы Seigolonhcet Erolagnab	~
~	Wipro Technologies, Bangalore 1	Огріш ,seigolonhceT Erolagnab	Огріы ,seigolonhceT Erolagnab	~

Passed all tests!

Finish review