

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
a b c	8
b c	

Answer: (penalty regime: 0 %)

```
1 import java.util.HashSet;
2 public class CommonAlphabets {
3     public static int getsingleDigitSum(int sum) {
4         while (sum >= 10) {
5             int tempSum = 0;
6             while (sum > 0) {
7                 tempSum += sum % 10;
8                 sum /= 10;
9             }
10            sum = tempSum;
11        }
12        return sum;
13    }
14    public static int findCommonAsciiSum(char[] input1, char[] input2) {
15        HashSet<Character> set1 = new HashSet<>();
16        for (char c : input1) {
17            set1.add(c);
18        }
19        int sum1 = 0;
20        for (char c : input2) {
21            if (set1.contains(c)) {
22                sum1 += (int) c;
23            }
24        }
25        return sum1;
26    }
27    public static void main(String[] args) {
28        char[] input1 = {'a', 'b', 'c'};
29        char[] input2 = {'b', 'c'};
30        int sum1 = findCommonAsciiSum(input1, input2);
31        int result = getsingledigitsum(sum1);
32        System.out.println(result);
33    }
34 }
```

	Input	Expected	Got	
✓	a b c	8	8	✓
	b c			

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 reversal 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.

NOTE:

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 Seigolonhct 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 erolagn
Wipro Technologies Bangalore 1	Orpiw Seigolonhct Erolagnab
Wipro Technologies, Bangalore 1	Orpiw ,seigolonhceT Erolagna

Answer: (penalty regime: 0 %)

```
1 import java.util.Scanner;
2 public class ReverseWordsWithCaseControl {
3     public static String reverseWordsWithCase(String sentence, int case_option) {
4         String[] words = sentence.split(" ");
5         StringBuilder result = new StringBuilder();
6         for (String word : words) {
7             String reversedWord = reverseWord(word);
8             if (case_option == 1) {
9                 reversedWord = reverseCaseWithOriginalPosition(reversedWord, word);
10            }
11            result.append(reversedWord).append(" ");
12        }
13        return result.toString().trim();
14    }
15    private static String reverseWord(String word) {
16        StringBuilder reversed = new StringBuilder(word);
17        return reversed.reverse().toString();
18    }
19    private static String reverseCaseWithOriginalPosition(String reversedWord, String originalWord) {
20        StringBuilder result = new StringBuilder(reversedWord);
21        for (int i = 0; i < originalWord.length(); i++) {
22            char originalchar = originalWord.charAt(i);
23            char reversedChar = reversedWord.charAt(i);
24            if (Character.isUpperCase(originalChar)) {
25                result.setCharAt(i, Character.toUpperCase(reversedChar));
26            } else if (Character.isLowerCase(originalchar)) {
27                result.setCharAt(i, Character.toLowerCase(reversedchar));
28            }
29        }
30        return result.toString();
31    }
32    public static void printResult(String input, int case_option) {
33        String result = reverseWordsWithCase(input, case_option);
34        System.out.println(result);
35    }
36    public static void main(String[] args) {
37        Scanner scanner = new Scanner(System.in);
38        String input = scanner.nextLine();
39        int case_option = scanner.nextInt();
40        printResult(input, case_option);
41        scanner.close();
42    }
}
```

Quiz navigation



Show one page at a time

[Finish review](#)

Status	Finished
Started	Tuesday, 12 November 2024, 12:22 PM
Completed	Tuesday, 12 November 2024, 12:55 PM
Duration	33 mins 25 secs

Question 1

Correc

Marked out of
5.00

 Flag question

You are provided with a string which has a sequence of 6 1's and 2's.

This challenge is the encoded version of a English word. You are supposed write a program to decode the provided string and find the original word.

This sequence is the encoded version of a English

Each alphabet is represented

THIS

2:0

Y:00

X:000

W : 0000

V : 00000

U : 000000

T : 0000000

and so on up