

Accenture Sections	Information	Questions and Time
Cognitive Ability	<ul style="list-style-type: none">• English Ability• Critical Thinking and Problem Solving• Abstract Reasoning	50 Ques in 50 mins
Technical Assessment	<ul style="list-style-type: none">• Common Application and MS Office• Pseudo Code• Fundamental of Networking, Security and Cloud	40 Ques in 40 mins
Coding Round	<ul style="list-style-type: none">• C• C++• Dot Net• JAVA• Python	2 Ques in 45 mins

DEBUG WITH SHUBHAM

Accenture Technical Assessment Detailed Overview

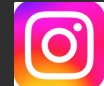
29-sep-2024 Coding Interview Questions



<https://www.youtube.com/@DebugWithShubham>



<https://www.linkedin.com/in/debugwithshubham/>



<https://www.instagram.com/debugwithshubham/>



<https://topmate.io/debugwithshubham>



<https://t.me/debugwithshubham>

Question-1

2. Hands On Programming

Question 1

How to Attempt?

You are a teacher organizing a field trip for your students. You have a class of **N** students, and you want to divide them into two groups for the trip. However, you have a special requirement: one group should only consist of students with even-numbered IDs, and the other group should only consist of students with odd-numbered IDs and both the groups must be of the same length. Your task is to find and return an integer value representing the maximum number of students that can be included in both groups.

Input Specification:

input1 : An integer value N representing the total number of students
input2 : An integer array containing the IDs of the students.

Output Specification:

Return an integer value representing the maximum number of students that can be included in both groups.

Example 1:

input1 : 4
input2 : {5,2,3,6}

Output : 2

Mettl Online Assessment © 2021-2031

Python

```
main.py +
1 def max_students_in_groups(n, ids):
2     even_count = 0
3     odd_count = 0
4     for student in ids:
5         if student % 2 == 0:
6             even_count += 1
7         else:
8             odd_count += 1
9     return min(even_count, odd_count)
10 n = 4
11 ids = [5, 2, 3, 6]
12 result = max_students_in_groups(n, ids)
13 print(result) # Output: 2
```

Main.java

JAVA

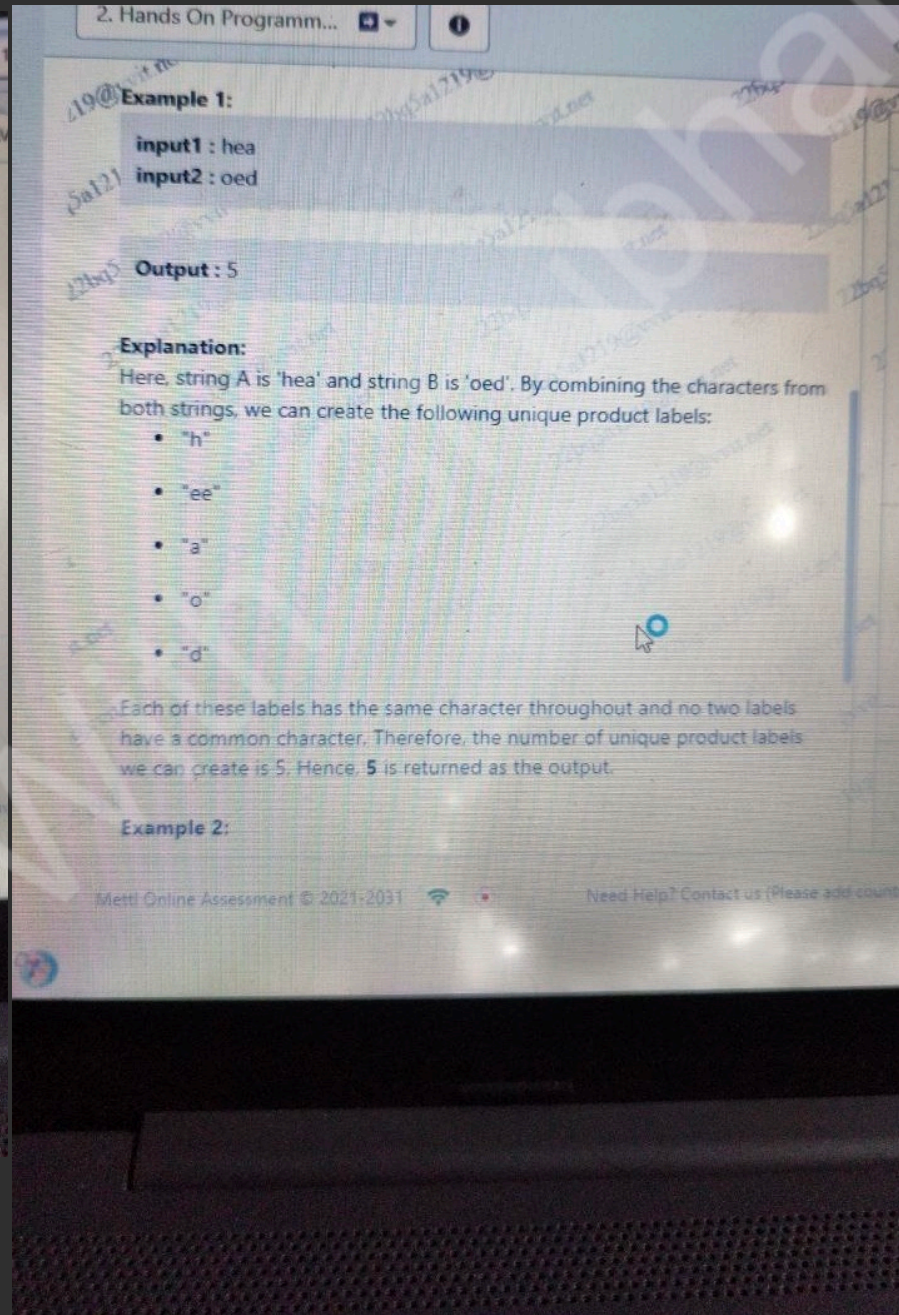
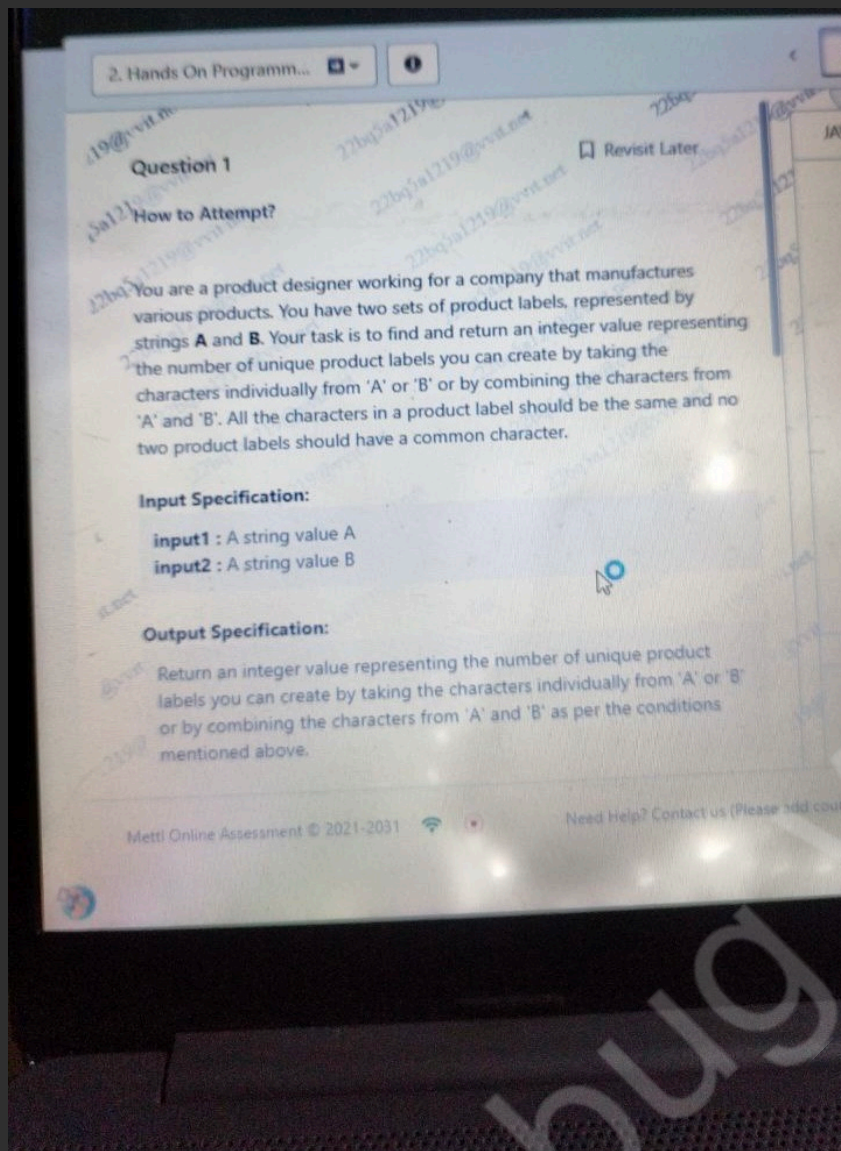
```
1 public class Main {
2     public static int maxStudentsInGroups(int n, int[] ids) {
3         int evenCount = 0;
4         int oddCount = 0;
5         for (int student : ids) {
6             if (student % 2 == 0) {
7                 evenCount++;
8             } else {
9                 oddCount++;
10            }
11        }
12        return Math.min(evenCount, oddCount);
13    }
14    public static void main(String[] args) {
15        int n = 4;
16        int[] ids = {5, 2, 3, 6};
17
18        int result = maxStudentsInGroups(n, ids);
19        System.out.println(result);
20    }
21 }
22
```

main.cpp

C++

```
1 #include <iostream>
2 #include <algorithm>
3 using namespace std;
4 int maxStudentsInGroups(int n, int ids[]) {
5     int evenCount = 0, oddCount = 0;
6     for (int i = 0; i < n; i++) {
7         if (ids[i] % 2 == 0) {
8             evenCount++;
9         } else {
10             oddCount++;
11         }
12     }
13     return min(evenCount, oddCount);
14 }
15
16 int main() {
17     int n = 4;
18     int ids[] = {5, 2, 3, 6};
19     int result = maxStudentsInGroups(n, ids);
20     cout << result << endl;
21     return 0;
22 }
23
```


Question-2



Brute-Force Solution

main.py +

```
1 def unique_product_labels(A: str, B: str) -> int:
2     set_A = set(A)
3     set_B = set(B)
4     unique_labels = set_A.union(set_B)
5     return len(unique_labels)
6 input1 = "heaa"
7 input2 = "oed"
8 output = unique_product_labels(input1, input2)
9 print(output) # Expected Output: 5
10
```

Main.java

```
1 import java.util.HashSet;
2 public class Main {
3     public static int uniqueProductLabels(String A, String B) {
4         HashSet<Character> setA = new HashSet<>();
5         HashSet<Character> setB = new HashSet<>();
6         for (char ch : A.toCharArray()) {
7             setA.add(ch);
8         }
9         for (char ch : B.toCharArray()) {
10             setB.add(ch);
11         }
12         setA.addAll(setB);
13         return setA.size();
14     }
15     public static void main(String[] args) {
16         String input1 = "heaa";
17         String input2 = "oed";
18         int output = uniqueProductLabels(input1, input2);
19         System.out.println(output);
20     }
21 }
22
```

main.cpp

```
1 #include <iostream>
2 #include <set>
3 using namespace std;
4
5 int uniqueProductLabels(string A, string B) {
6     set<char> setA(A.begin(), A.end());
7     set<char> setB(B.begin(), B.end());
8     setA.insert(setB.begin(), setB.end());
9     return setA.size();
10 }
11 int main() {
12     string input1 = "heaa";
13     string input2 = "oed";
14     int output = uniqueProductLabels(input1, input2);
15     cout << output << endl;
16     return 0;
17 }
18
```


Python

```
main.py +
1 def unique_product_labels_v2(A: str, B: str) -> int:
2     freq = [0] * 26
3     for char in A:
4         freq[ord(char) - ord('a')] = 1
5     for char in B:
6         freq[ord(char) - ord('a')] = 1
7     return sum(freq)
8
9 input1 = "abc"
10 input2 = "def"
11 output = unique_product_labels_v2(input1, input2)
12 print(output)
```

JAVA

Main.java

```
1 public class Main {
2     public static int uniqueProductLabelsV2(String A, String B) {
3         int[] freq = new int[26];
4         for (char ch : A.toCharArray()) {
5             freq[ch - 'a'] = 1;
6         }
7         for (char ch : B.toCharArray()) {
8             freq[ch - 'a'] = 1;
9         }
10        int sum = 0;
11        for (int count : freq) {
12            sum += count;
13        }
14        return sum;
15    }
16    public static void main(String[] args) {
17        String input1 = "abc";
18        String input2 = "def";
19        int output = uniqueProductLabelsV2(input1, input2);
20        System.out.println(output);
21    }
22 }
23 }
```

Optimal Case Solution

C++

main.cpp

```
1 #include <iostream>
2 #include <string>
3 using namespace std;
4 int uniqueProductLabelsV2(const string& A, const string& B) {
5     int freq[26] = {0};
6     for (char ch : A) {
7         freq[ch - 'a'] = 1;
8     }
9     for (char ch : B) {
10        freq[ch - 'a'] = 1;
11    }
12    int sum = 0;
13    for (int i = 0; i < 26; i++) {
14        sum += freq[i];
15    }
16    return sum;
17 }
18 int main() {
19     string input1 = "abc";
20     string input2 = "def";
21     int output = uniqueProductLabelsV2(input1, input2);
22     cout << output << endl;
23     return 0;
24 }
25 }
```