**Experiment No. - 1.3**

**Student Name: Deepak Saini UID: 20BCS4066**

**Branch: 20BCC1 Section/Group: A**

**Semester:**  **5th Date of Performance: 17/08/2022**

**Subject Name: ADVANCED PROGRAMMING LAB**

**Subject Code: 20CSP-334**

1. **Aim/Overview of the practical:**

You are given a string containing characters “A” and “B” only, your task is to

change it into a string such that there are no adjacent matching characters, to do

this , you are allowed to delete 0 or more characters in the string, your task is to

find the minimum number of required deletions.

1. **Task to be done:**

You are given a string containing characters “A” and “B” only, your task is to change it into a string such that there are no adjacent matching characters, to do this , you are allowed to delete 0 or more characters in the string, your task is to find the minimum number of required deletions.

1. **Steps for practical**:
2. Include the header files.
3. Take the string as input.
4. Make a count variable to count to count the number of deletions required.
5. Iterate through the string’s each character through for loop , and compare , adjacent elements, if they are same then increase the count variable.
6. Finally, output the minimum number of characters required to be deleted, by printing count variable value.

**4 . Code:**

#include<bits/stdc++.h>

using namespace std;

int32\_t main()

{

int test;

cout<<"Enter number of test cases: "<<endl;

cin>>test;

while(test--)

{

string str;

cout<<"Enter the strings: "<<endl;

cin>>str;

int count=0;

for(int i=1; i<str.length(); i++)

{

if(str[i]==str[i-1])

{

count++;

}

}

cout<<"The minimum number of characters required to be deleted: "<<count<<endl;

}

return 0;

}

**5.** **Output:**



