Experiment Number 3

Name ::   Rishabh Anand   UID ::   19BCS4525

Branch ::   CSE - IoT   Sec/Grp ::   1/A

Semester ::   5th Date ::   12th Sept, 2021

Subject ::   Adv Programming Lab   CODE ::   CSP-347

1. Aim :

Find the minimum number of required deletions in a string of 'A' and 'B' such that there are no matching adjacent characters.

2. Task :

1. Find the number of required deletions in the string s.t. there are no matching adj. characters

3. Algorithm :

1. Take a input string

2. Iterate through the string to find all repeating chracters.

3. Increment counter if found.

4. Print the counter variable.

4. Source Code :

```cpp
#include <bits/stdc++.h>

// using namespace std;

int main()
{
    int t;
    std::cin >> t;

    while (t--)
    {
        std::string s;

        int c = 0, a = 0;
        std::cin >> s;

        for (int i = 1; s[i] != '\0'; i-=-1)
        {
            if ((s[i] == 65 && s[a] == 66) ||
                (s[i] == 66 && s[a] == 65))
                a = i;
            else
                c++;
        }

        std::cout << c << std::endl;
    }
    return 0;
}
```

5. Observations :

```
base  master  2?  $  g++ stringMatching.cpp
5
AAAA
3
BBBBB
4
ABABABAB
0
BABABA
0
AAABBB
4
```

Learning Outcomes :

- Strings
- String Matching
- String Alteration

S. No.	Parameters	Marks Obtained	Maximum Marks
1.			
2.			
3.			