



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Experiment 7 (String)

Student Name: Rajdeep Jaiswal

UID: 20BCS2761

Branch: BE CSE

Section/Group: 902 b

Semester: 5th

Subject Name: Competitive Coding

Subject Code: 20CSP_314

1. Aim/Overview of the Practical:

- a. Strong Password.
- b. Camelcase.

2. Task to be done / Which logistics used:

- a. Louise joined a social networking site to stay in touch with her friends. The signup page required her to input a name and a password. However, the password must be strong. The website considers a password to be strong if it satisfies the following criteria:

Its length is at least 6.

It contains at least one digit.

It contains at least one lowercase English character.

It contains at least one uppercase English character.

It contains at least one special character. The special characters are:

!@#\$%^&*()-+

She typed a random string of length n in the password field but wasn't sure if it was strong. Given the string she typed, can you find the minimum number of characters she must add to make her password strong?

- b. There is a sequence of words in CamelCase as a string of letters, , having the following properties:



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

It is a concatenation of one or more words consisting of English letters.

All letters in the first word are lowercase.

For each of the subsequent words, the first letter is uppercase and rest of the letters are lowercase. Given s , determine the number of words in s .

Steps for experiment/practical/Code:

a. Strong Password:

```
#include <bits/stdc++.h> using namespace
```

```
std;
```

```
int check(string s, string w) {  
    for (char c : s) {  
        if (w.find(c) != string::npos) {  
            return 0;  
        }  
    }  
    return  
    1;  
}
```

```
int main() {  
    int n; cin  
    >> n; string  
    s;  
    cin >> s;  
    string numbers = "0123456789";  
    string lower_case = "abcdefghijklmnopqrstvwxyz"; string  
    upper_case = "ABCDEFGHIJKLMNOPQRSTUVWXYZ";
```



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

```
string special_characters = "!@#$%^&*()-+"; int
ans = 0; ans += check(s, numbers); ans +=
check(s, lower_case); ans
+= check(s, upper_case); ans
+= check(s,
special_characters); ans =
max(ans, 6 - n); printf("%d\n",
ans); return 0;
}
```

Camel Case:

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

```
using namespace std; int
main(){ string s; cin >> s;
int t=1; for (int
i=0;i<s.length();i++) if
(isupper(s[i])) t++;
cout<<t<<endl;
return 0;
}
```

Result/Output/Writing Summary:

a. Strong Password:



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

A screenshot of a testing interface. On the left, there is a vertical list of test cases from 'Test case 0' to 'Test case 6', each with a green checkmark and a lock icon. The main area on the right is titled 'Compiler Message' and contains a 'Success' message. Below this, there is a section for 'Input (stdin)' with two lines of input: '3' and 'Ab1'. To the right of this section is a 'Download' link. Below the input section is a section for 'Expected Output' with one line of output: '3'. To the right of this section is another 'Download' link.

b. Camel Case:



DEPARTMENT OF

Discover. Learn. Empower.

COMPUTER SCIENCE & ENGINEERING

A screenshot of a code editor interface. On the left, there is a list of test cases from 'Test case 0' to 'Test case 6', each with a green checkmark and a lock icon. The main area shows the output for 'Test case 0'. It includes a 'Compiler Message' section with the text 'Success'. Below that is the 'Input (stdin)' section, which contains a single line of code: `saveChangesInTheEditor`. To the right of this code is a 'Download' link. The 'Expected Output' section shows a single line of text: `5`, also with a 'Download' link to its right.

Learning outcomes (What I have learnt):

- Learnt about strings.
- Got an overview of the implementation of strings.
- Get to know about crucial test cases.