Assignment 3 - DSA - UEC613

Ishaan Bhola - 102015051 - 2NC6

Q1. Write a function to perform following operations on the string: (Note: You can make single function for all operations/independent function for each problem)

i) Finding length of a string

Function:-

```
private static int stringlength(String str){
    int count=0;
    for(char ch: str.toCharArray()){
        count++;
    }
    return count;
}
```

Output:-

```
String length = 3
```

Input = “abc”

ii) Converting a string in lowercase

Function:-

```
private static String stringLowerCase(String str){
    String newStr="";
    for(int i=0;i<str.length();i++){
        char ch = str.charAt(i);
        if(ch>='a' && ch<='z'){
            newStr+=ch;
        }else{
            newStr+=(char)(ch+32);
        }
    }
    return newStr;
}
```

Output:-

```
String to lower case = abc
```

Input=”ABC”

iii) Counting number of words and vowels in a string

Function:-

```
    private static void wordsAndVowels(String str){
        int vowels=0;
        int words=0;

        for(int i=0;i<str.length();i++){
            char ch = str.charAt(i);
            if(ch=='a' || ch=='e' || ch=='i' || ch=='o' || ch=='u'
|| ch=='A' || ch=='E' || ch=='I' || ch=='O' || ch=='U'){
                vowels++;
            }
            if(ch==' ') words++;
        }
        System.out.println("Vowels = " + vowels + "  Words=
"+(words+1));
    }
```

Output:-

```
Vowels = 1  Words= 1
```

Input=”abc”

iv)Validating a string

Function:-

```
    private static boolean isValidString(String str){

        for(int i=0;i<str.length();i++){
            char ch = str.charAt(i);
            if(!(ch>='A' && ch<='z')){
                return false;
            }
        }
        return true;
    }
```

Output:-

```
Is string valid? true
```

Input=”abc”

v) Reversing a string
Function:-

```
private static String reverseString(String str){
    String newStr = "";
    for(int i=str.length()-1;i>=0;i--){
        newStr+=str.charAt(i);
    }
    return newStr;
}
```

Output:-

```
Reversed string = cba
```

Input=”abc”

vi)Checking if a string is palindrome
Function:-

```
private static boolean palindrome(String str){
    int i=0,j=str.length()-1;

    while(i<j){
        if(str.charAt(i)!=str.charAt(j)){
            return false;
        }
        i++;
        j--;
    }
    return true;
}
```

Output:-

```
Is string palindrome? false
```

Input = “abc”

vii)Finding duplicate characters in a string (Note: print the duplicate characters only once, irrespective of the number of times it occurred)x

Function:-

```
    private static void duplicateChars(String str){
        int chars[]=new int[256];
        for(char ch:str.toCharArray()){
            chars[ch]++;
        }
        for(int i=0;i<256;i++){
            if(chars[i]>1) System.out.print((char)i +" ");
        }
    }
```

Output:-

```
Duplicate chars in string = c d
```

Input= “abcccdd”

Q2) Sort the characters of the string entered by user.

Function:-

```
    private static void sortString(String str) {
        char ch[]= str.toCharArray();
        for(int i=0;i<ch.length;i++){
            for(int j=1;j<ch.length;j++){
                if(ch[j]<ch[j-1]){
                    char temp = (char)ch[j-1];
                    ch[j-1]=(char)ch[j];
                    ch[j]=(char)temp;
                }
            }
        }
        for(char c:ch){
            System.out.print(c);
        }
    }
```

Output:-

1st string is input while 2nd is output

```
zcedoepsnqalf
acdeeflnopqsz
```

Q3) Implement combination formula nCr using recursion

Functions:-

```
    private static int factorial(int n){
        if(n==1) return 1;
        return n*factorial(n-1);
    }
    private static int NcR(int n,int r){
        int num = factorial(n);
        int denom = factorial(n-r)*factorial(r);
        return num/denom;
    }
```

Output:-

NcR when n=5 and r=2

```
10
```

Q4) Implement the Fibonacci series using recursion

Functions:-

```
    private static int fibonnaci(int n){
        if(n<=1) return n;
        return fibonnaci(n-1)+fibonnaci(n-2);
    }
```

Output:-

```
55
```

For n = 10

Q5) Write a program for finding the factorial of a number recursively.

Function:-

```
    private static int factorial(int n){
        if(n==1) return 1;
        return n*factorial(n-1);
    }
```

Output:-

```
120
```

For n= 5