Lab Manual – String Instructions

Important Instructions:

- **Make proper subroutines**
- **Use Delay Function, that we did in class (if required)**

Activity 1 [MOVS]: Write a program that copies first 12 lines of screen (e.g. DOSBOX Introduction) in last 12 lines.

Activity 2 [Scroll Up and Down / Screen Oscillation]: We did examples of Scroll Up and Scroll Down in class. Using those functions, write a program that scrolls the Screen 3 rows Up and 3 rows down in an infinite loop without losing any data of screen. (First call printstr to print 5,5 strings in top and bottom 5 rows of screen to test your ScrollUpAndDown functionality.) (Do it with both local and global arrays).

```
abc: times 32 dw 0 ; space for 32 words
xyz: times 256 dw 0 ; space for 256 words
```

Practice Problems

1- Write a program that finds total number of occurrences of a character from a null terminated string. For example total occurrences of 'a' in string "I am a student of coal" are 3.

2- Write a program that prints tokens of a string on Screen. Best use string instructions studied so far.

Sample Run:

```
String: I am a student of coal
Output:
I
am
a
student
of
coal
```

3- [SCAS] Write a program that takes a c-string *myStr* and two characters *charToFind* and *charToReplace* from user and replaces all the occurrences of *charToFind* with *charToReplace* in *myStr*. Your program should create a space of 50 characters on heap in order to save *myStr*.

Sample output:

InputString: ddsdfhgrtsdfhjghjksdd

CharToFind: d

CharToReplace: $

ModifiedString: $$s$fhgrts$fhjghjks$$

4- Write a program that takes a character *ch* and a CString *myStr* and removes all the occurrences of *ch* from *myStr*.

Sample Output:

---
myStr: cabccdefcfdcxyzcc

ch: 'c'

Modified String: abdeffd xyz

5- TrimStart(char* str)

Write a function that takes a string and removes all the space in start of the string.

Sample Output:

Before TrimStart

str: "   Hello   How are you?"

After TrimStart

str: "Hello   How are you?"

6- String Compression

Write a function that compresses a string by removing consecutive occurrences of same character.

Sample Run:

String Before Compression:

Str: "ggggdddddddyyyyakxxxuww"

String after Compression:

Str: "gdyakxuw"

7- Write a function that searches a substring from a string and highlights the found substring. If the string is not found it will not highlight anything.

Sample Run:

String: "I am a student of COAL"

Substring: "student"

Printed String after Search: "I am a student of COAL"