# CS1160
# Lab 10: Strings

## What are Strings?

A string is a new data type that is used to store multiple characters or text.

It is defined as a sequence of characters that are terminated with a null character \0. The string can be also considered as an array of character.

## Strings Declaration and Initialization

```
char string_name[size];
```

**Example:**

```
char a[6] = "Hello";   or   char b[ ] = "Hello";
```

## Reading and Printing Strings

use function "**gets**". This function reads a line of string, until it meets newline entered from the keyboard.

And, to print the string value; use the function **"puts"**.

```
char Name[ ];
gets  (Name);
puts  (Name);
```



Sara Abu Sa'aleek

## Iterating Strings

To iterate within a string you need to know its size/length, and use the null character to identify the end of the string and terminate the loop.

**Example 1: Count the occurrences of 'a' in the string**

```c
#include <stdio.h>

int main()
{
   char s[100]; // Allocate enough space for user input
    int i = 0, count = 0;

    printf("Enter a string: ");
    gets(s); // Read string from user

    puts("The string is: ");
    puts(s);
    while (s[i] != '\0') { // Loop through the string
        if (s[i] == 'a' || s[i] == 'A') {
            count++;
        }
        i++;
    }

    printf("The number of A & a: %d\n", count);

    return 0;
}
```

```
Enter a string: Sara Ahmad
The string is:
Sara Ahmad
The number of A & a: 4
```

## ASCII values in C

Each character variable is represented in numerical numbers called **ASCII**. The ASCII codes for the **lowercase letters 'a' through 'z' are 97 to 122.** The codes for **'A' through 'Z' are 65 to 90**.

**Example 2: Count the number of capital letters in the string**

```c
#include <stdio.h>

int main() {
    int i, uppercount = 0;
    char w[100]; // Allocate enough space for user input

    printf("Enter a string: ");
    gets(w); // Read string from user

    for (i = 0; w[i] != '\0'; i++) {
        if (w[i] >= 'A' && w[i] <= 'Z') {
            uppercount++;
        }
    }

    printf("The number of uppercase letters: %d\n", uppercount);
    return 0;
}
```

```
Enter a string: Sara AHMAD
The number of uppercase letters: 6
```

## Passing Strings to Functions

Strings can be passed to a function in a similar way as arrays.

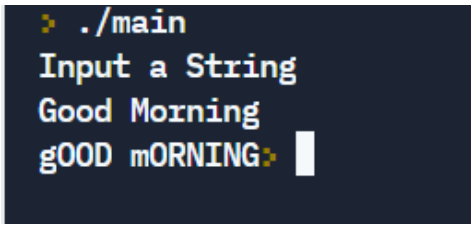
**Example 3:**

```c
#include <stdio.h>

int length(char s[]) {
    int i;
    for (i = 0; s[i] != '\0'; i++);
    return i;
}

int main() {
    char st1[100]; // Allocate enough space for user input

    printf("Enter a string: ");
    gets(st1); // Read string from user

    int c = length(st1); // function call
    printf("The length is: %d\n", c);

    return 0;
}
```

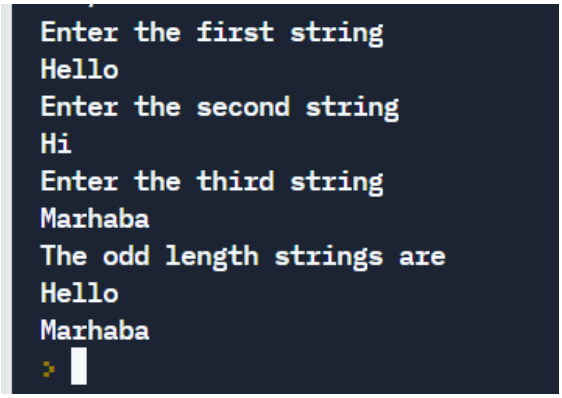
```
Enter a string: Sara Ahmad
The length is: 10
```

## Tasks

1. Write a C program that reads a string then toggles this string (converts all uppercase letters to lowercase letters, and converts all lowercase letters to uppercase letters).
   **Sample Output**




2. Write a C program that reads three strings then, calls a user-defined function to find out the length of each string. The function will print out the strings with odd length ONLY.
   **Sample Output**




3. Write a C program that reads a string then it will calls a function that will replace all occurrences of the character “e” with “3”.
   **Sample Output**