Faculty of Computing

CS110: Fundamentals of Computer Programming

Class: BESE-16B

Lab 12: Text Processing Using Strings

CLO 2	Solve given real-world problem by applying appropriate programming concepts and techniques.
CLO 3	Build a program and associated documentation using appropriate IDE and supplementary tools.

Date: 2nd December 2025

Time: 2:00 pm-5:00 pm

Instructor: Dr. Momina Moetesum

Lab Engineer: Mr. Nadeem Nawaz

Lab 12: Text Processing Using Strings

Learning Objectives:

After completing this section, you will be able to:

- Use Character Arrays and Strings to solve various text-based problems.
- Use member functions of <string> class for string manipulation.
- Use C-string functions for C-string manipulation.

Practice Tasks

(a) Write a C++ program to test the following code snippets. Include the appropriate headerfiles.

```
string message = "ABCD";
cout << message.length() << endl;
cout << message.at(0) << endl;
string s = "Bottom";
cout << s.size() << endl;
cout << s.at(1) << endl;
string final = message + s;
```

```
string s1 = "Good morning";
string s2 = "Good afternoon";
cout << s1[0] << endl;
cout << (s1 == s2 ? "true": "false") << endl;
cout << (s1 != s2 ? "true": "false") << endl;
cout << (s1 > s2 ? "true": "false") << endl;
cout << (s1 >= s2 ? "true": "false") << endl;
cout << (s1 < s2 ? "true": "false") << endl;
cout << (s1 <= s2 ? "true": "false") << endl;
```

Task 1: [CLO 2]

Using C++ Style Strings and <string.h> functions, write a program that prompts the user to enter three cities and displays them in ascending order. Here is a sample run:

```
Enter the first city: Chicago
Enter the second city: Los Angeles
Enter the third city: Atlanta

The three cities in alphabetical order are Atlanta Chicago Los Angeles
```

Task 2: [CLO 3]

C-string is an array of characters that ends with the null terminator character '\0'.
char city1[] = "Dallas"; // C-string
char city2[] = {'D', 'a', 'l', 'l', 'a', 's'}; // Not a C-string
You can process C-strings using <cstring.h> functions in the C++ library. Some of these are given below:

Function	*Description*
`size_t strlen(char s[])`	Returns the length of the string, i.e., the number of the characters before the null terminator.
`strcpy(char s1[], const char s2[])`	Copies string s2 to string s1.
`strncpy(char s1[], const char s2[], size_t n)`	Copies the first n characters from string s2 to string s1.
`strcat(char s1[], const char s2[])`	Appends string s2 to s1.
`strncat(char s1[], const char s2[], size_t n)`	Appends the first n characters from string s2 to s1.
`int strcmp(char s1[], const char s2[])`	Returns a value greater than 0, 0, or less than 0 if s1 is greater than, equal to, or less than s2 based on the numeric code of the characters.
`int strncmp(char s1[], const char s2[], size_t n)`	Same as strcmp, but compares up to n number of characters in s1 with those in s2.
`int atoi(char s[])`	Returns an int value for the string.
`double atof(char s[])`	Returns a double value for the string.
`long atol(char s[])`	Returns a long value for the string.
`void itoa(int value, char s[], int radix)`	Obtains an integer value to a string based on specified radix.

ISBN-13 is a standard for identifying books.

It uses 13 digits *d*1*d*2*d*3*d*4*d*5*d*6*d*7*d*8*d*9*d*10*d*11*d*12*d*13. The last digit *d*13 is a checksum, which is calculated from the other digits using the following formula:

10 - (d1 + 3d2 + d3 + 3d4 + d5 + 3d6 + d7 + 3d8 + d9 + 3d10 + d11+ 3d12)%10

If the checksum is 10, replace it with 0.

Using a C-style string (rather than a string) for storing the ISBN numbers. Write the following function that obtains the checksum from the first 12 digits:

int getChecksum(const char s[])

Test the function using a main driver.

Deliverables:

Compile a single Word document with codes for each question and screenshots of the outputs and submit this Word file on LMS.

Lab Rubrics

Your Lab 12 will be graded out of 5 for each rubric according to the following rubrics. Grades for CLO3 will be shared at different intervals during the semester.

Lab Rubrics for Lab 10 (Two-dimensional Arrays)

Sr. No.	Assessment	Unacceptable (0 Marks)	Does Not Meet Expectations (1/2 Marks)	Meets Expectations (3/4 Marks)	Exceeds Expectations (5 Marks)
1	**Application of Programming Concepts** **(CLO2, PLO3)**	The student did not submit any work. OR The student plagiarized the solution and/or used unfair means.	The student is unable to apply the appropriate programming concepts to solve the given problem thus resulting in an incomplete or ineffective solution. The program flow is messy and incomprehensible. Codes are non-modular and cannot be reused.	The student requires some guidance to apply the appropriate programming concepts to solve the given problem. The program flow requires minor improvements. Codes are semi-modular and semi-reusable.	The student demonstrates a clear ability to apply the appropriate programming concepts to solve the given problem. The program flow is adequate. Codes are modular, reusable, and easily readable.
2	**Software Tool Usage** **(CLO3-PLO5)**		The student demonstrates a lack of understanding of tool usage. Implementation has syntax/semantic/runtime errors, and the student is unable to debug and correct the errors. The code has inadequate comments and variable names and does not adhere to the coding standards. No Error handling has been performed. Documentation is poorly structured.	The student demonstrates some understanding of tool usage. The codes are correct in terms of their syntax, however, the program output is not always correct in all test cases. The code has limited comments and inconsistent variable names and may not adhere to the coding standards. Some Error handling has been performed. Documentation is adequately structured.	The student demonstrates a good understanding of tool usage. Furthermore, his/her coding is complete and functional, and the program output is correct in all test cases. The code has sufficient comments and consistent variable names and reasonably adhere to the coding standards. Adequate Error handling has been performed. Documentation is well structured.