National University of Computer and Emerging Sciences

Laboratory Manual

for

Data Structures Lab

Course Instructor	Dr Amna Khan
Lab Instructor(s)	Muhammad Saddam Mian Bassam Ahmad
Section	CS-B
Date	08-Oct-2020
Semester	FALL 2020

Department of Computer Science

FAST-NU, Lahore, Pakistan

Objectives:

In this lab, students will practice:

1. Stack Implementation using linked List
2. Stack Application

Problem Description

Design a C++ program that use a stack for matching tags and quotes in HTML-Hyper Text Markup Language. Your program will get an HTML code in an input file and it should figure out if tags and quotes are properly matched or not using stack. In case the tags are not properly matched then your program should report following

i) the first error **ii)** print the mismatched tag and
iii) inform the line number where the starting tag occurred.

What is HTML?

HTML is a language for describing web pages.

- HTML stands for **H**yper **T**ext **M**arkup **L**anguage
- HTML is not a programming language, it is a **markup language**
- The purpose of the tags is to **describe page content**

HTML ELEMENTS

- An HTML element starts with a **start tag / opening tag**
- An HTML element ends with an **end tag / closing tag**
- The **element content** is everything between the start and the end tag
- Some HTML elements have **empty content**
- Most HTML elements can have **attributes**

```
<!DOCTYPE html>
<html>
<head >
    <title>My Web Page</title>
</head>
<body>
    <form id="form1">
        <div id="div1">
            This is Text...
        </div>
    </form>
</body>
</html>
```

In above example
<!DOCTYPE html>

The header should come in the start of the document and it will be always same.

<html>, <head>, <title>, <body>, <form> and are tags and each must have a corresponding ending tags </html>, </head>, </title>, </body>, </form> and .

Your program should handle the following features of HTML:

1. **HTML (header), it will come as it is.**
2. **HTML tags. Make it case sensitive.**
3. **HTML attribute**
4. **HTML comments, start with <!-- and ends with -->** for example: <!--this is comment in html--> **Consider another example**

```
<!DOCTYPE html>
<html>
<head>
    <!--this is comment in html-->
    <title>My Web Page</title>
</head>
<body>
    <text id="Text1" name="TextBox1"> </text>
    <button id="btn1" type="submit">  Submit Button  </button>
</body>
</html>
```

In the example above:
<title> and <button>have **text content** because they contain text ("**My Web Page**" and "**Submit Button**").

<text> has empty content between opening and closing tag while <title> and <button> has text content.
<text> and <button> has an **attribute** (id="Text"), there can be more than one attribute in one tag.

Note that Attribute values must always be quoted. Only double quotes can be used.

Your program should keep track that attributes have opening and closing quotes.

Your code will have template-based Node class and template-based Stack class. Implement stack using singly linked list.

```
template <class
T> class Node
{
public:
        Node()
        T data;
        Node<T> * next;


};
```

```
template<class T>
class Stack {
public:
              Stack() ;
              ~Stack();
               bool IsEmpty();
               bool push(const T &
val);               bool pop(T & val);
              T top();
               void print();

private:
                Node<T> * top;
};
```

Create htmlData class
Think about its attributes carefully

```
Void main(){

   Stack<htmlData> S1;
}
```

CODE DESIGN GUIDLEINES

- Do template-based programming
- Code should be properly intendent and commented (2 marks for this)
- Make sure there are no memory leaks or dangling pointers