

# National University of Computer and Emerging Sciences, Lahore Campus



Course:	Data Structures	Course Code:	CS 2001
Program:	BS(Computer Science)	Semester:	Fall 2021
Due Date	06-Nov-2021 at 11:59 pm	Total Marks:	25
Type:	Assignment 2	Page(s):	3

## Important Instructions:

1. You have to submit the solution of both questions separately. There will be two submission folders on Google classroom, one for question 1 and the other for question 2.
  - a. For question 1, name your solution file with your roll number, i.e., 20\_1111.cpp, and submit in question 1's submission folder on Google classroom. Do not zip your file.
  - b. For question 2, name your solution file with your roll number, i.e., 20\_1111.cpp, and submit it in question 2's submission folder on Google classroom. Do not zip your file.
2. You are not allowed to copy solutions from other students. We will check your code for plagiarism using plagiarism checkers. If any sort of cheating is found, negative marks will be given to all students involved.
3. Late submission of your solution is not allowed.

## Question 1: [10 marks]

HTML is a standard markup language for Web pages. It describes the structure of a Webpage. HTML consists of a series of elements which tell the browser how to display the content. HTML elements are represented by tags for example <html>, <head> etc. **Each tag has a corresponding closing tag.** Web browsers do not display the HTML tags, but use them to render the content of the page. Consider a following simple html document.

```
<html>

<head>
<title>Page Title</title>
</head>

<body> <h1>This is a Heading</h1>

<p>This is a paragraph.</p>

</body>

</html>
```

Now write a program that checks whether an html file is well-formed or not, i.e., each opening tag has a corresponding closing tag or not. Also, the examples of html files that are not well formed are:

```
<html>

<head>
<title>Page Title</title>
</head>

<body> <h1>This is a Heading</h1>

<p>This is a paragraph.</p>

</body>
</p>
</html>
```

```
<html>

<head>
<title>Page Title</title>
</head>

<body> <h1>This is a Heading</h1>

<p>This is a paragraph.</p>

</html>
```

The following html document is also not well-formed, although there is a closing tag for every opening tag. Your program should be able to handle this.

```
<html>

<head>
<title>Page Title
</head>
</title>
<body> <h1>This is a Heading</h1>

<p>This is a paragraph.</p>
<\body>
</html>
```

There are some other rules as well in html, such as a title tag always comes inside the head tag block. There is no need to handle this rule.

**Hint:** Use stack for this problem. You can make use of STL stack. You can only use the following functions of STL stack: push, pop, top, and empty.

## Question 2: [15 marks]

### A simple version of snakes and ladder

Implement a game that initially asks how many players will play the game. The game will then ask the name of each player. The game is played as follows:

1. The game is played iteratively. In every iteration, every player takes turn to roll a dice. The face value of dice is added to the total score of the player.
2. The first player to reach 100 score gets 1<sup>st</sup> position. The second player to reach 100 score gets 2<sup>nd</sup> position.
3. When a player reaches 100, he no longer remains part of the play. If current score + face value is greater than 100, then the player will retry in the next iteration, i.e., the score must exactly become 100.
4. There are ladders at 20 and 60. If a player's score becomes exactly 20, the score becomes 40. If a player's score is exactly 60, the score becomes 75. Similarly, there are snakes at 50 and 90. If a player's score is exactly 50, the score becomes 40. If a player's score is exactly 90, the score becomes 79.
5. When n-1 players have reached 100, the game ends and the position of all players that participated in the game is displayed in ascending order.
6. During game play, some useful messages should be displayed, so that one can know what is going on in the game, such as:

Please enter the number of players: 2

Please enter the name of Player 1: ABC

Please enter the name of Player 2: XYZ

It is ABC's turn, please enter any key to continue.

ABC got 5. The total score of ABC is now 5.

It is XYZ's turn, please enter any key to continue.

XYZ got 6. The total score of XYZ is now 6.

It is ABC's turn, please enter any key to continue.

ABC got 3. The total score of ABC is now 8.

.

.

.

It is XYZ's turn, please enter any key to continue.

XYZ got 4. The total score of XYZ is 50. There is a snake at 50, so total score is now 40.

Use queue for this problem. You can use the queue class available in STL. You can only use push, pop, front, and empty functions of STL queue. Apart from queues, you cannot use any other data structures including arrays in any way (not even for storing the record of players that have already finished the game).