CPTS_223 Advanced Data Structures in C++
Micro-assignment 2: BSTs

I. Learner Objectives:

At the conclusion of this programming assignment, participants should be able to:

- Implement C++ code from within the constraint of another person's code base
- Understand logics of binary search trees.

II. Prerequisites:

Before starting this programming assignment, participants should be able to: Analyze a basic set of requirements and apply top-down design principles for a problem

- Describe and analyze binary search trees (BSTs)
- Design, implement, and test medium programs in an object-oriented language
- Edit, build, and run programs through a Linux environment

III. Overview & Requirements:

In this micro-assignment, you will be implementing some missing code for a class BST template provided. You must work within your Linux environment. Please use the project given on Canvas. Specifically, you will need to implement code for the following 4 functions in BST.h and make sure all existing test cases in main() have the correct outputs. The only file you need to change is BST.h.

- (25 pts)~BST destructor
- (25 pts) contains
- (25 pts) printLevelOrder
- (25 pts) printMaxPath

Currently, the output of the test cases in main() is

```
-10 -1 1 6 11 100
TODO: Implement printLevelOrder
100 in BST? true (1) or false (0): TODO: Implement contains
1
9 in BST? true (1) or false (0): TODO: Implement contains
1
Nodes count: 6
Height: 3
Max path: TODO: Implement printMaxPath

After 11 removed: -10 -1 1 6 100
TODO: Implement printLevelOrder
Nodes count: 5
TODO: Implement Destructor
```

The correct output of the test cases in main() should be

```
-10 -1 1 6 11 100
11
1 100
-1 6
-10
100 in BST? true (1) or false (0): 1
9 in BST? true (1) or false (0): 0
Nodes count: 6
Height: 3
Max path: 11 1 -1 -10
After 11 removed: -10 -1 1 6 100
100
1
-1 6
-10
Nodes count: 5
```

IV. Submitting Assignments:

1. Option 1: Canvas:
   Zip all source files into a single zip file, and upload it on Canvas.
2. Option 2: Git:
   1) On your local file system, and inside of your Git repo for the class, create a new branch called MA2, push all files of the given project to the MA2 branch of your private GitHub repo created in PA1.
   2) Your project must build properly by GitHub Action.
   3) Submission: You must submit a URL link to the branch of your private GitHub repository. Please make sure the instructor and TAs (GitHub accounts listed in Syllabus) are the collaborators of your repository. Otherwise, we won't be able to see your repository. DO NOT CREATE NEW REPO.
   4) Do not push new commits the branch after you submit your URL to Canvas otherwise it might be considered as late submission.

V. Grading Guidelines:

This assignment is worth 100 points.