BST Number List in Console/Terminal
Cpt S 321 Homework Assignment
Washington State University

Submission Instructions:

- Create a GitLab repository on the EECS server that you will use for all the assignments in this class.
- Add me and the TAs as part of your team (see syllabus for user ids).
- Tag the version that you would want us to correct with the assignment number. For example, "HW1", "HW2", etc. Feel free to create a branch for each homework assignment.
- Your project must contain a readme file where you will write your full name and WSU ID.

Assignment Instructions:

Read all the instructions *carefully* before you write any code.

Create a C# console application that fulfills the following requirements:

1. Get a list of integer numbers from the user on A SINGLE LINE
    - The numbers will be in the range [0,100]
    - The numbers will be separated by spaces
    - You may assume that the user enters a correctly formatted input string that meets these requirements
    - You may use Console.ReadLine or a similar method to get input from the user
2. Add all the numbers to a binary search tree in the order they were entered
    - Don't allow duplicates
    - Use the Split function on the entered string for easy parsing (split on the space character)
3. Display the numbers in sorted order (smallest first, largest last).
    - Traverse the tree in order to produce this output.
4. Display the following statistics about the tree
    - Number of items (note that this will be less than or equal to the number of items entered by the user, since duplicates won't be added to the tree). Write a function that determines this from your BST, NOT the array returned from the split. In other words, you must have a Count function in your BST implementation.
    - Number of levels in the tree. A tree with no nodes at all has 0 levels. A tree with a single node has 1 level. A tree with 2 nodes has 2 levels. A tree with three nodes could have 2 or 3 levels. You should know why this is from your advanced data structures prerequisite course.

- Theoretical minimum number of levels that the tree could have given the number of nodes it contains (figure out the formula to calculate this)

Point Breakdown (10 points total):

- 6 points: Fulfill all the requirements above with no inaccuracies in the output and no crashes.
- 1 point: Code is clean, efficient and well organized.
- 1 point: Quality of identifiers.
- 1 point: Existence and quality of comments.
- 1 point: Existence and quality of test cases.

Sample Output: