Week 6 - Forest for the Trees

Submit Assignment

Due No Due Date **Points** 18 **Submitting** a text entry box or a file upload

Assignment Requirements

1. Generate a random list of integers.
   Show the binary heap tree resulting from inserting the integers on the list one at a time.
2. Using the list from the previous question, show the binary search tree resulting from the list as a parameter to the buildHeap method.
   Show both the tree and list form.
3. (optional bonus question) Extend the `buildParseTree` function to handle mathematical expressions that do not have spaces between every character.
4. (optional bonus question) Extend the `buildParseTree` and evaluate functions to handle boolean statements. Remember that "not" is a unary operator, so this will complicate your code somewhat.

Please appropriately name your files to reflect which question they answer.

In addition to coding these tasks, you must post a video running and explaining your code. This allows for you to demonstrate what is occurring in the code as it is happening and how it is organized. You must also run your code in the video to explain the output and why the program produced that output.

GitHub Submission Instructions

Once you complete these exercises, be sure you have committed your solutions locally and pushed them up to the remote repository. If you are unsure how to clone the repository for this assignment, please review **Pull and Push for Assignments**.

Canvas Submission Instructions

When you have completed this assignment and pushed your work to the remote GitHub repository, answer the following question(s):

1. How many hours do you estimate you used completing this assignment?
2. What was easiest for you when completing this assignment?
3. What was the most difficult challenge you experienced when completing this assignment?

To begin, click the Submit Assignment button in Canvas and respond in the available text entry box.

Tree Search Rubric

Criteria	Ratings			Pts
Generate list of integers	**1.0 pts** **Full Marks**	**0.0 pts** **No Marks**		1.0 pts
Binary Heap Tree Readability Code is readable and well organized	**3.0 pts** **Full Marks**	**1.0 pts** **Partial**	**0.0 pts** **No Marks**	3.0 pts
Binary Tree Output Output shows binary heap tree from inserting integers one at a time.	**3.0 pts** **Full Marks**	**1.0 pts** **Partial**	**0.0 pts** **No Marks**	3.0 pts
buildHeap method Use Users the buildHeap method properly	**2.0 pts** **Full Marks**	**0.0 pts** **No Marks**		2.0 pts
Part 2 Output Proper output is produced from using the buildHeap method	**3.0 pts** **Full Marks**	**1.0 pts** **Partial**	**0.0 pts** **No Marks**	3.0 pts
Answer all Canvas questions	**1.0 pts** **Full Marks**	**0.0 pts** **No Marks**		1.0 pts
Video Submission Clearly explains the organization and output of the program.	**5.0 pts** **Full Marks**	**3.0 pts** **Partial**	**0.0 pts** **No Marks**	5.0 pts

Total Points: 18.0