

**Due: 11:59pm, Thursday, Oct. 19, 2023**

## Learning Objectives

The goal of this assignment is to gain experience with binary trees.

## Instructions

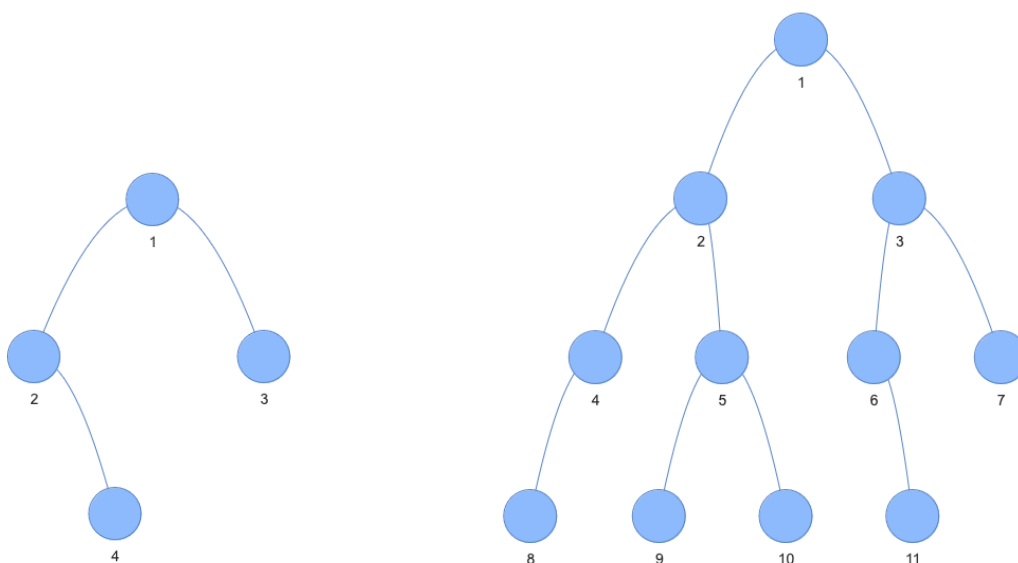
Answer the questions using the `BinaryTree` class in the provided Python source file. Do not modify any existing methods in the `BinaryTree` class.

The `BinaryTree` class uses an array to implement a binary tree following the indexing scheme presented in the lectures. The root node is stored in `self._data[1]`, and the left and right child nodes for any node in position `self._data[k]` are stored in `self._data[2*k]` and `self._data[2*k+1]`. The first element of the array is not used, and `None` is always stored in `self._data[0]`.

## Question 1: (35 pts)

Implement the method named `read_string(s)` in the `BinaryTree` class. This method creates a binary tree given its string representation, `s`.

Consider the following binary trees:



Example string representations of these two trees are:

- `s = '1, 2, 3, None, 4'`
- `s = '1, 2, 3, None, 4, None, None, None, None, None, None'`
- `s = '1, 2, 3, 4, 5, 6, 7, 8, None, 9, 10, None, 11, None, None'`

You can assume that `s` is a valid string representation of a binary tree. You do not need to worry about handling incorrectly formatted strings. However, your method should work for any string that is a valid binary tree representation.

## Question 2: (65 pts)

In the lectures, we studied preorder, postorder and inorder traversals. Methods for these traversals are included in the `BinaryTree` class that is provided.

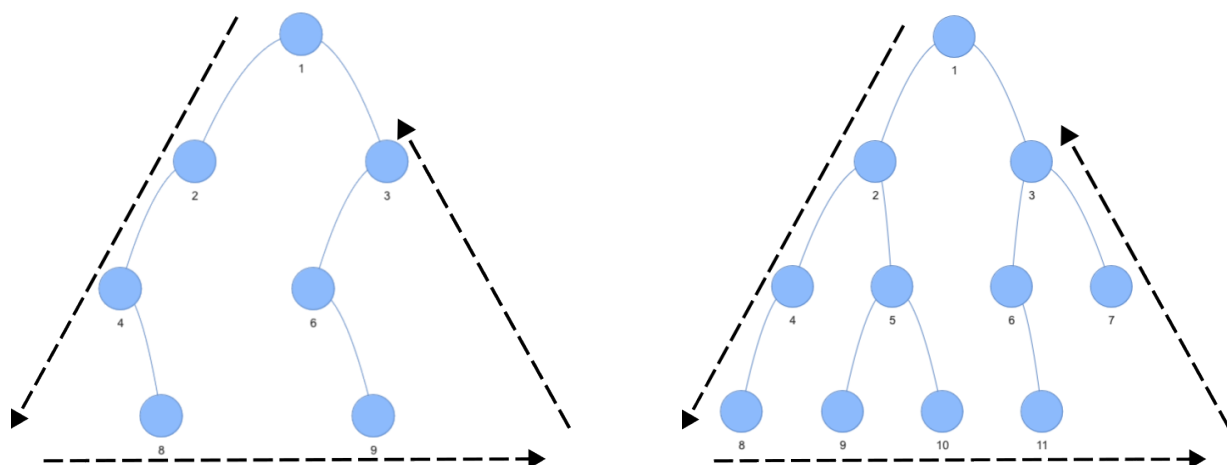
Implement the `boundary_traversal()` method. This method prints only the nodes on the exterior boundaries of the binary tree.

A boundary traversal starts at the root node, printing all of the leftmost nodes, then all of the exterior nodes along the bottom level of the tree, followed by all of the rightmost nodes. One way to think about this traversal is to consider the binary tree as enclosed by an equilateral triangle. All the nodes that are adjacent to the triangle's edges are printed.

For example, `boundary_traversal()` for the two following trees would have outputs:

Boundary Traversal: `['1', '2', '4', '8', '9', '3']`

Boundary Traversal: `['1', '2', '4', '8', '9', '10', '11', '7', '3']`



## Submission

Submit a single zip file containing the Python source file with your modifications to the `BinaryTree` class through the Assignment submission folder in Brightspace.

Python source code should be `*.py` plain text. The only file types allowed aside from Python source code (`*.py`) are pdfs and plain text (`*.txt`). Do not submit Word documents or rich text format documents. They will not be marked. Only submit a single zip archive. Do not submit files archived in rar format. That may result in your assignment not being graded.

Name all files with the format “firstname\_lastname\_studentid...”. Make sure to include your name and student ID as comments at the top of all Python source files.

Late submissions will be subject to a 10% penalty for each hour past the deadline.

## Attribution

Submissions must represent your independent work.

If your submitted work includes unacknowledged collaboration, code materials, ideas or other elements that are not your original work, it may be considered plagiarism or some other form of cheating under MUN general regulations 6.12.4.2 and academic penalties will be applied accordingly.

The submission of work that has been created by generative artificial intelligence (GAI) tools and presented as a student’s original work is considered an academic offence in this course. Using AI tools without proper citation constitutes plagiarism, and your work will be subject to the appropriate Memorial’s Academic Misconduct policy.

If your submission contains any contribution by others, including internet sources and classmates, then you should include an attribution section detailing the extent of these contributions. This will also help distinguish what elements of the submission are original. You may not receive full credit if your original elements are insufficient, but can lessen penalties for plagiarism or copying if you acknowledge your sources.

## Github

I encourage you to store and version your work on Github. It is good practice to do so as everyone uses git in the real world.

However, **it is a requirement that git repositories containing assignment material be private.** University regulations section 6.12.4.2 consider it cheating if you allow your work to be copied. There will be zero tolerance for this.