Data Structures

Lab Exercise 9 (Binary Search Tree)

Task 1:   **(Points 1)**

Download `lab9_startercode.cpp` from NYU classes/Brightspace and complete following methods of BST (Binary Search Tree) class.

a) `Node* getRoot()`   (0.1 Point)
   A method which returns a pointer to the root of the Tree.

b) `int height(Node *ptr)`   (0.1 Point)
   A method which finds the Height (Maximum Depth) of a tree/subtree rooted at `ptr`.

c) `Node* findMin(Node *ptr)`   (0.1 Point)
   A method which finds and returns a pointer to the Node with minimum value of `key` in a tree/subtree rooted at `ptr`.

d) `Node* findMax(Node *ptr)`   (0.1 Point)
   A method which finds and returns a pointer to the Node with maximum value of key in a tree/subtree rooted at `ptr`.

e) `Node* findKey(Node *ptr, int key);`   (0.1 Point)
   A method which finds and returns a pointer to the Node with key value in a tree/subtree rooted at `ptr`.

f) `void insert(Node* ptr, int key)`   (0.2 Points)
   A method which inserts a Node with `key` in a tree/subtree rooted at `ptr`.

g) `Node* remove(Node *ptr, int key);`   (0.2 Point)
   A method which removes a node with `key` from the tree/subtree rooted at `ptr`.

h) `void printInorder(Node* ptr)`   (0.1 Point)
   A method which traverses (in-order) and prints the `keys` of a tree/subtree rooted at `ptr`.

that there is never a valid reason to share your code with fellow students, and that there is no valid reason to publish your code online in any form.

Every student is responsible for the work they submit. If there is any doubt during the grading about whether a student created the assignment themselves (e.g. if the solution matches that of others), we reserve the option to let the student explain why this is the case. In case doubts remain, or we decide to directly escalate the issue, the suspected violations will be reported to the academic administration according to the policies of NYU Abu Dhabi.

(see https://students.nyuad.nyu.edu/campus-life/community-standards/policies/academic-integrity/)