

## CS241 – Data Organization Fall 2020

### Programming Assignment #9: Recursive Binary Tree Functions Due by 11:59 PM on Tuesday, November 24th

In this class we have seen how the basic depth-first-search, tree-hugging traversal of a Binary Tree can be written very compactly as a recursive function. By augmenting this basic traversal algorithm, we can perform a variety of different computations on a binary tree. For this assignment you must implement the following binary tree functions.

```
bool  isHeapOrdered ( Node * curr );  
  
int   numSingleChild ( Node * curr );  
  
void  makeMirror ( Node * curr );
```

The provided source code contains a **stub** for each of these functions, together with a description of what the function should do. You must complete the **body** of each of these functions.

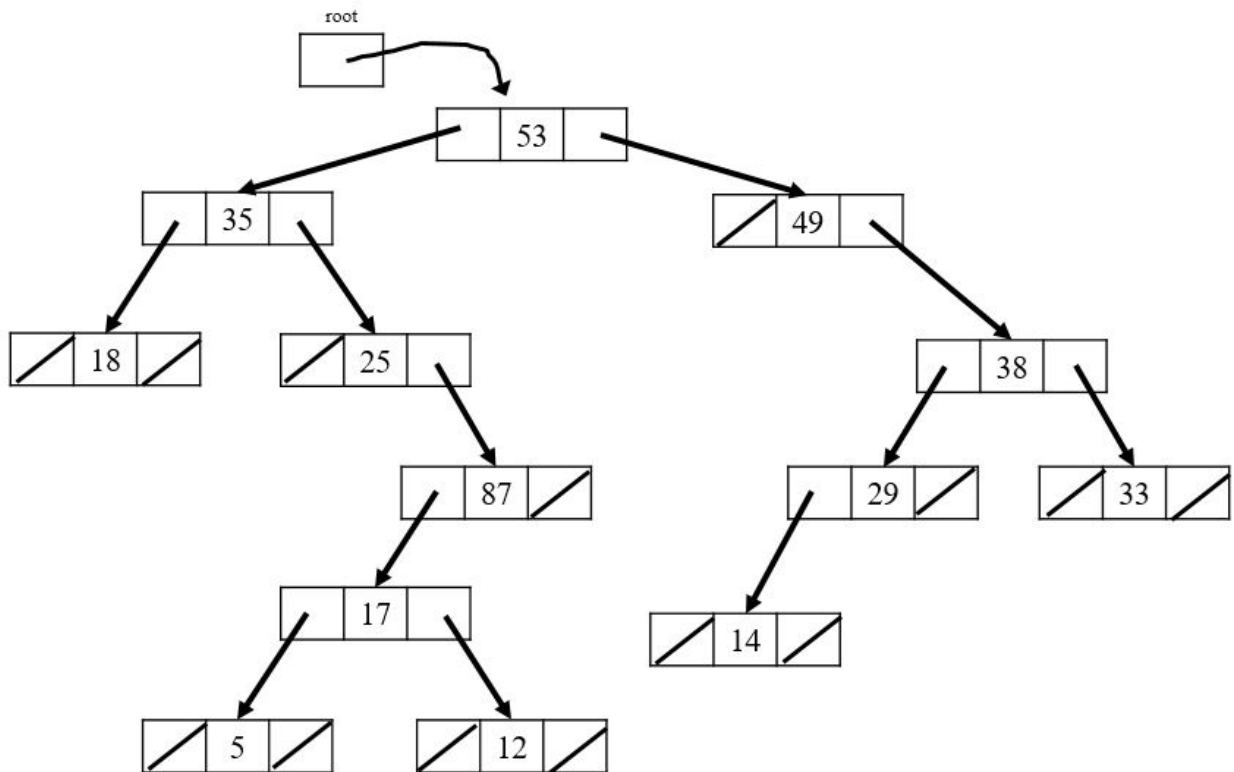
The **isHeapOrdered** function returns true or false depending on whether the binary tree is max-heap-ordered (i.e., if every node's value is greater-than-or-equal-to the values of its children). Note that we are **not** concerned at all about the shape of the tree. A tree can be heap-ordered even if it does not have a "complete" shape.

The **numSingleChild** function returns the number of nodes in the tree that have exactly one child. For example, in the tree below, the function would return the value 4.

The **makeMirror** function alters the shape of the tree (ie, it alters the left and right child pointers) so that the tree is now a mirror-image of the former shape. In other words, the tree has been "flipped" through an imaginary vertical line through the root of the tree.

The given source code contains the two functions **size** and **contains** that we have studied in this class.

The main function creates and initializes the following binary tree that is used as a “test case”.



You should implement each of these functions in the **most efficient manner** possible.

You should **modify** the code at: `~soraya/cs241_Soraya/programs/RBTF/pa9_rbt.c`

You should be sure to include **THE NAME OF THE AUTHOR OF THE PROGRAM** in a comment at the top of your source code file, for this and all other assignments in this course.

Your source code must use proper **style**, that is variables should be well named (name is not too short, not too long, and is meaningful), and bodies of loops, if's, etc.. should be properly indented. Refer to the coding style file for this class, published on Learn under Coding Standards in a file named: `cs241_codingStandards2020.pdf`.

Create a `.c` file for this assignment and name it using your last name and the initial of your first name, like this: **lastName\_initialFirstName\_RBTF.c**. Submit this file for grading to Learn in the place of this assignment.