// Keaton Spiller

// CS302

// 03/05/2024 Winter 2024

// Program # 3

Debugger Writeup

As I was inserting one node at a time into the 2-3 tree the debugger allowed me to display what values were in the current node, and the nodes children. This was useful for seeing where the tree went, and what I needed to change to get the tree to meet each edge case.

This helped me solve the first big split where the tree is at level 2 and must split off the middle node from both subtrees. Originally when inserting 7 into my tree in the below example I was only splitting on the right subtree, and after using gdb I was able to trace where the value 7 goes as it’s inserted.

In this particular case it showed going to the right subtree, because 7 >= 4, and on the right subtree 7 >= 6, therefore I rotated 6, 5 and 7 into the right subtree, yet was able to see where I then need to throw the root back into the tree in a separate function which goes through and rotates the left subtree to account for the new level.

The reason print statements didn’t work for seeing this example, was the print statements used the numbers from my display height function to see each level. The levels would incorrectly print the height of unbalanced trees, and 1, 3 data values in the left subtree node would incorrectly show levels of 3, which made debugging using testing confusing and too difficult.  
  
e.g.)  
Before gdb:  
After Inserting (7): [2,4] level 1  
 [1,3] [6,] level 2  
 [5,] [7,] level 3  
After gdb:  
Before Inserting (7): [2,4] After Inserting (7): [4,] level 1  
 [1,][3,][5,6] [2,] [6,] level 2  
 [1,] [3,] [5,] [7,] level 3

gdb helps glass box testing, when we know the inner workings of the program, and we are looking inside to fix input, and output.

I would prefer to know how to step through gdb in a cleaner manner, so when I have multiple display prints for each step through the code I can easily see what each is printing, without having to use ctrl+L to clear, so I can manage the visibility on my small laptop screen a bit better.