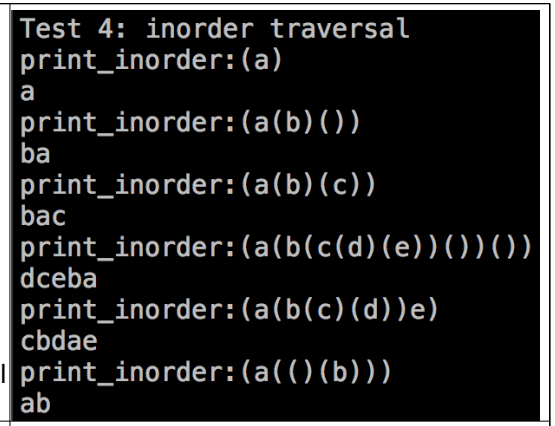
#	Pri.	Story	Example	Test
1		As a Programmer, I want to print a tree so I can see what is in it.	If you type “trees -p '(a(b()())(c()()))' you will see (a(b##)(c##)).	Test 1: print_tree Empty test tree: # Test tree (a(b)(c)): (a(b##)(c##))
2		As a Programmer, I want a function to print a pre-order traversal of a complete binary tree, to see how it is done.	If you type “trees -e '(a(b()())(c()()))' you will see abc.	Test 2: preorder traversal print_preorder:(a) a print_preorder:(a(b)()) ab print_preorder:(a(b)(c)) abc print_preorder:(a(b(c(d)(e))()))()) abcde print_preorder:(a(b(c)(d))e) abcde print_preorder:(a(()(b))) ab
3		As a Programmer, I want a function to print a post-order traversal of a complete binary tree, to see how it is done.	If you type “trees -o '(a(b()())(c()()))' you will see bca.	Test 3: postorder traversal print_postorder:(a) a print_postorder:(a(b)()) ba print_postorder:(a(b)(c)) bca print_postorder:(a(b(c(d)(e))()))()) decba print_postorder:(a(b(c)(d))e) cdbea print_postorder:(a(()(b))) ba

4		As a Programmer, I want a function to print a in-order traversal of a complete binary tree, to see how it is done.	If you type “trees -i '(a(b()())(c()()))' you will see bac.	
6		As a Programmer, I want to add a node to a binary search tree, to see how it is done.	If you type “trees -b 'abc' you will see (a#(b#(c##))).