

BSTreeInsert

Your task is to write a function, `BSTreeInsert`, that inserts a given value into a BST, if it does not already exist in the BST, and returns the root of the updated BST. The insertion should be performed using normal leaf insertion. Do not rebalance the tree. If the value already exists in the BST, the function should do nothing.

Download

Click [here](#) to download a zip of the files.

The Files

| | |
|---------------------------|---|
| BSTree.c | Contains code for reading and printing a BST |
| BSTree.h | Contains the definition of the BST data structure and function prototypes |
| testBSTreeInsert.c | Contains the main function, which reads in a BST and the value to be inserted from standard input, calls <code>BSTreeInsert</code> , and prints out the result. |
| BSTreeInsert.c | Contains <code>BSTreeInsert</code> , the function you must implement |
| Makefile | A makefile to compile your code |
| tests/ | A directory containing the inputs and expected outputs for some basic tests |
| autotest | A script that uses the tests in the tests directory to autotest your solution. You should only run this after you have tested your solution manually. |

Examples

Your program should behave like these examples:

```
$ ./testBSTreeInsert
Enter the preorder traversal of the BST: 6 3 1 4 8 10
Enter value to insert: 7
BST before inserting 7:

      6
     / \
    /   \
   3     8
  / \   / \
 1  4 10

BST after inserting 7:

      6
     / \
    /   \
   3     8
  / \   / \
 1  4 7 10
```

```
$ ./testBSTreeInsert
Enter the preorder traversal of the BST: 7 5 3 1
Enter value to insert: 4
BST before inserting 4:

      7
     /
    5
   /
  3
 /
1

BST after inserting 4:

      7
     /
    5
   /
  3
 / \
1  4
```

```
$ ./testBSTreeInsert
Enter the preorder traversal of the BST:
Enter value to insert: 5
BST before inserting 5:
```

X

BST after inserting 5:

5

```
$ ./testBSTreeInsert
Enter the preorder traversal of the BST: 1 9 2 8
Enter value to insert: 5
BST before inserting 5:
```

```
1
 \
  9
 /
2
 \
 8
```

BST after inserting 5:

```
1
 \
  9
 /
2
 \
 8
 /
5
```

Testing

You can test your program manually by compiling your code using **make**, and then running **./testBSTreeInsert**, as shown above. After you are satisfied with your solution, you can autotest it by running **./autotest**. This will run some basic tests on your program.