instructions.md

A6

Due: See dropbox details
Submit: .zip

Details

For this assignment you will be taking the code you wrote for A5 and be adding the following function prototype and implementations to your program.

```
bool remove(int);
int findHeight();
node* findMin();
node* findMax();
```

The above prototype must be added to your `bst.h` file and their implementations must be added to `bst.cpp`. As before you can (and likely need to) create additional functions which will help support the new functionality.

To start, copy your A5 files and add the new prototypes to `bst.h` and begin writing your implementations in `bst.cpp`. I have provided the following updated files to get you going

- `main.cpp` Updated to reflect the new functions
- `cmd.txt` Updated to reflect the new functions
- `output.txt` Updated to reflect the new functions

How I will test your program

To test your program I will use your `bst.h` and `bst.cpp` files and combine them with my own `main.cpp` file. My version of `main.cpp` will be similiar to the one that is provided with a few extra grading features and test cases added in.

Deliverables

Add `bst.h`, `bst.cpp`, `main.cpp`, `makefile`, and `cmd.txt` to a zip file called `a6.zip` and submit that file to the dropbox. I will only accept submissions that meet these requirements.

Expections

- No use of any prebuilt tree libraries
- Your code should be well formatted, points may be taken for sloppy code
- Your output should match the output provided in output.txt
- You must submit a zip file called `a5.zip`
- Your code should be fairly robust and be able to handle obvious edge cases such as searching for a number which is not in the tree.

Special behavior

There is a unique case to think about when dealing with a BST. That is what happens if we try to insert a number that is already in the tree into the tree again? It is expected that only one occurance of a number or node should ever appear in a tree, so if you ever try to insert the same node into a tree twice then your program should just not insert the duplicate node into the tree.