```
/*
Filename: p5.h
Author(s): Zachary Rea and Parker Ross
Date: 19 February 2023
Description: The header file for p5
*/
#ifndef __P5_H
#define __P5_H

#include <string>
//*****************************************************************************
class sAVL;
class sNode {
    private:
        //Node key
        std::string text;
        //For future use
        int h;
        //Pointers for left and right children
        sNode *left, *right;
        //Constructor for the node
        sNode(std::string text = "");
        friend sAVL;
};
class sAVL {
    private:
        //Current size of the tree
        int treeCount;
        //Points to the top of the tree of nodes
        sNode *root;
        //Function to find the minimum value of the subtree
        std::string findMin(sNode *ptr);
        //Function for help with recursion
        bool insert(sNode *&p, std::string text);
        //Function for help with recursion
        bool remove(sNode *&p, std::string text);
        //Function for help with recursion
        bool isIn(sNode *p, std::string text) const;
        //Function for help with recursion
        void printIt(sNode *p, int &index) const;
        //Function for help with recursion
        void clear(sNode *p);
        //Function for a left rotate
        void rotateLeft(sNode *&p1);
        //Function for a right rotate
        void rotateRight(sNode *&p1);
        //Function to balance a node
        void bal(sNode *&p);
        //Function to return the height of a node
        int height(sNode *p) const;
        //Function to calculate the height of a node based on children
        int calcHeight(sNode *p);
    public:
        //Constructor
        sAVL();
        //De-constructor
        ~sAVL();
        //Function to insert the text into the tree
        bool insert(std::string text);
        //Function to remove the node with the given text
        bool remove(std::string text);
        //Function to tell if the tree contains the text
        bool isIn(std::string text) const;
        //Function to print the BST values in ascending order
        void printIt() const;
        //Function to show the number of nodes in the tree
        int count() const;
        //Function to remove all nodes from the tree
        void clear();
```

```
70    };
71    #endif
```