

```

1  /*
2  Filename: p4.h
3  Author(s): Zachary Rea and Parker Ross
4  Date: 12 February 2023
5  Description: The header file for p4
6  */
7  #ifndef __P4_H
8  #define __P4_H
9
10 #include <string>
11 //*****
12 class sBST;
13 class sNode {
14     private:
15         //Node key
16         std::string text;
17         //For future use
18         int h;
19         //Pointers for left and right children
20         sNode *left, *right;
21         //Constructor for the node
22         sNode(std::string text = "");
23         friend sBST;
24 };
25 class sBST {
26     private:
27         //Current size of the tree
28         int treeCount;
29         //Points to the top of the tree of nodes
30         sNode *root;
31         //Function to find the minimum value of the subtree
32         std::string findMin(sNode *ptr);
33         //Function for help with recursion
34         bool insert(sNode *p, std::string text);
35         //Function for help with recursion
36         bool remove(sNode *p, std::string text);
37         //Function for help with recursion
38         bool isIn(sNode *p, std::string text) const;
39         //Function for help with recursion
40         void printIt(sNode *p) const;
41         //Function for help with recursion
42         void clear(sNode *p);
43     public:
44         //Constructor
45         sBST();
46         //De-constructor
47         ~sBST();
48         //Function to insert the text into the tree
49         bool insert(std::string text);
50         //Function to remove the node with the given text
51         bool remove(std::string text);
52         //Function to tell if the tree contains the text
53         bool isIn(std::string text) const;
54         //Function to print the BST values in ascending order
55         void printIt() const;
56         //Function to show the number of nodes in the tree
57         int count() const;
58         //Function to remove all nodes from the tree
59         void clear();
60 };
61 #endif

```