

## 1) (10 pts) DSN (Binary Search Trees)

Complete writing the function shown below recursively, so that it takes in a pointer to the root of a binary search tree of strings, *root*, and a string, *target*, and returns 1 if the string is contained in the binary search tree and false otherwise. You may assume all strings stored in the tree contain lowercase letters only. In order to receive full credit, your function must run in  $O(h)$  time, where  $h$  is the height of the binary search tree storing all of the words.

```
typedef struct bstNode {
    struct bstNode *left, *right;
    char str[100];
} bstNode;

int search(bstNode *root, char* target){

    if (root == NULL) return 0;                // 2 pts
    int cmp = strcmp(target, root->str);

    if (cmp < 0)                                // 1 pt
        return search(root->left, target);    // 2 pts
    else if (cmp > 0)                            // 1 pt
        return search(root->right, target);    // 2 pts
    return 1;                                  // 2 pts
}
```