

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

Write a function named *fsl()* (which stands for “find smallest least”) that takes a pointer to the root of a binary tree as its only argument and returns the value of the smallest leaf node in the tree. Note that the tree passed to your function will not necessarily be a binary search tree. If the pointer root is NULL, fsl should return INT\_MAX, which is defined below.

You cannot write any helper functions for this problem. You must complete all of your work in a single function. The function signature and node struct are given below.

```
#define INT_MAX 2147483647

typedef struct node {
    int data;
    struct node *left;
    struct node *right;
} node;

int fsl(node *root) {

    int l_min;
    int r_min;

    if (root == NULL)        // 2 pts: checking for NULL as base case
        return INT_MAX;    // 1 pt

    if (root->left == NULL && root->right == NULL) // 2 pts: identify leaf
        return root->data; // 1 pt: correct return
                           // value when leaf
                           // is encountered

    l_min = fsl(root->left); // 2 pts: correct recursive calls (give
    r_min = fsl(root->right); // only 1 pt here if only one recursive
                           // call)

    return (l_min < r_min) ? l_min : r_min; // 2 pts: returning min of
                                           // these two values

}
```