

Quiz #3 Answers

1. Write code that will find the predecessor of a node in a sorted binary tree. Assume this is a private only method so it deals directly with nodes.

```
Node *getPredecessor(Node *cur) {
    Node *ret=0;
    if(cur->left!=0) {
        ret=cur->left;
        while(ret->right!=0) {
            ret=ret->right;
        }
    } else {
        ret=cur->parent;
        while(ret!=0 && cur==ret->left) {
            cur=ret;
            ret=cur->parent;
        }
    }
    return ret;
}
```

2. How do we maintain some form of balance in an AVL tree? What are the requirements for a tree to be considered an AVL tree?

The only real requirement on an AVL tree is that the heights of the two children of a node can not differ by more than 1. This requirement along keeps the tree well balanced.

Extra Credit: What is the maximum length of any single path in a red-black tree? $2 \cdot \log_2 n$