4) (10 pts) DSN (Binary Trees)

Mark and his buddy Travis have devised a password scheme to secure files that they send among themselves. Their scheme hides the password in a string of English letters. The password is the alphabetically ordered sequence of the consonants in the string. So as not to have to compute the password each time, Mark has written a function called `printPassword`, which takes the letters of the original string stored in a binary search tree and prints out the password. For example, if the string in the message is **mental**, the password printed out would be **lmnt**. Or if the string was ***fragile***, then the password would be ***fglr.*** You may call the following function in your solution:

```
// Returns 1 if c is a consonant, 0 otherwise.
int isConsonant(char c)
```

Using the struct definition given below, complete the function in the space provided.

```
typedef struct treenode {
    char ch;
    struct treenode *left;
    struct treenode *right;
} treenode;

void printPassword(treenode* root) {

    if (root != NULL) {                    // 2 pts
        printPassword(root->left);         // 2 pts
        if (isConsonant(root->ch))         // 2 pts
            printf("%c", root->ch);        // 2 pts
        printPassword(root->right);        // 2 pts
    }
}
```