3) (10 pts) DSN (Linked Lists)

Write a **recursive** function, `aboveThreshold`, that takes in a pointer to the front of a linked list storing integers, and an integer, `limit`, and returns the number of values stored in the linked list that are strictly greater than `limit`. For example, if the function was called on a list storing 3, 8, 8, 6, 7, 5, 7, 9 and `limit` equaled 6, then the function should return 5, since the 2nd, 3rd, 5th, 7th and 8th values in the list are strictly greater than 6. (Notice that we don't count the 4th element.)

Use the struct definition provided below.

```
typedef struct node {
    int value;
    struct node* next;
} node;

int aboveThreshold(node* front, int limit) {

    if (front == NULL) return 0;

    int cnt = 0;
    if (front->data > limit) cnt = 1;

    return cnt + aboveThreshold(front->next, limit);
}
```

Grading conceptually: 3 pts for the base case, 3 pts for adding 1 in the case that the first item is greater than the limit, 3 pts for adding in the appropriate recursive call, 1 pt for returning. Max of 3 pts for an iterative solution.