Matt Muldowney
Homework 8 due 4/24

Written Problems

Problem 1

```
bool is_linear(qnode *hd) {
    if (hd == NULL) return true; //return true if nothing is queued

    qnode *n1 = hd; //slow pointer
    qnode *n2 = hd; //fast pointer

    //traverse queue until one pointer reaches the end of the queue
    while (n1 != NULL && n2 != NULL && n2->next != NULL) {
        n1 = n1->next;
        n2 = n2->next->next;

        if (n1 == n2) {
            return false; //return false if pointers point to same node
        }
    }

    return true; //return true if either pointer reaches end of queue
}
```

The **space cost is O(1)** because only a constant amount of extra memory (namely, the two pointers) in addition to the linked-list itself is allocated.

The **time cost is O(n).** There are three terminating conditions for the while loop. The first case is when n1 traverses the entire linked-list until it reaches the end, which has linear time cost. The second case is when n2 traverses half of the linked-list until it reaches the end, which also has linear time cost. Either of these cases will occur if the linked-list is in fact linear. If the linked-list is not linear, the third case occurs. In this case, the two pointers loop back to previous points in the linked-list until the fast pointer, n2, catches up to the slow pointer, n1. Even if either pointer loops through the linked-list multiple times, the total number of steps will still be a constant multiple of n in the worst case which would make it an element of O(n). Thus, no matter the terminating case, the time cost is linear.