Exercises for lab 8 of CS2211b

Instructor: Marc Moreno Maza, TA: Changbo Chen

March 13, 2011

In *Lecture 25: Abstract Data Types in C*, we discussed in detail how to implement a linked list. Recall that a linked list is defined as follows

```
struct nodestr {
  int data;
  struct nodestr *next;
};
typedef nodestr node;
```

In this lab, you are asked to implement an operation called `remove` to remove an element from a linked list. You need to implement this operation in two different ways:

1. Directly remove an element from the list, which implies that the list will be modified directly. The prototype of this function is

   ```
   node *remove1(node *head, int d);
   ```

   This function returns a pointer which points to the list.

2. Remove an element from the list without modifying the list, which means that you create a new list, which is the same as original list except the given element is removed from the new list. The original list will not be modified. The prototype of it is

   ```
   node *remove2(node *head, int d);
   ```

This function returns a pointer pointing to the new list.

To test your two functions, you are also required to

1. write a function called `print_list`, which will print the elements of a list.

2. modify the main program in our lecture 25 in order to demonstrate that the two remove functions work (call `print_list` to test).