CS102 – Review for exam over Chapter 7

Vocabulary to know

Pointers ☺

Pointer Basics

Know how to declare a pointer variable that points to a given data type
Know how to assign a pointer variable to the address of a regular variable using &
Know how to dereference a pointer
Be able to use an assigned pointer value
For example:

```
    int num;
    int* int_ptr =  &num;
    *int_ptr = 13;
```

Pointers not in use should be set to NULL
Know which variables below are pointer variables and which are not

```
        int *pointer1, pointer2, pointer3
```

Know the difference between the functions below

```
    void withdraw (double& balance, double amount)
    void withdraw (double* balance, double amount)
```

1. They are called differently
   You call the first function with
   ```
       withdraw (harrys_account, 100);
   ```
   and the second with
   ```
       withdraw (&harrys_account, 100);
   ```
2. It’s a question of who does the work—the compiler or you

Pointer arithmetic and the Array/Pointer Duality Law

The law says that an array variable can be treated as a pointer to the array
A pointer can be declared that steps through the array as the pointer is incremented/decremented
Be able to do pointer arithmetic along the lines of the array/pointer duality law
An example is

```
    int counts [10];
    int *p = counts;
    for (int i=0; i<10; i++)
    {
        cout << *p << endl;
        p++;
    }
```

More advanced use of ++, --

Be able to trace code using ++, -- as in

```
    x = y++;
```

By “trace”, I mean tell what is stored in x, or, tell what’s printed by: cout <<x;

structs

Be able to create a struct and use it (in printing, assigning, an if, etc.)

Linked lists

Know how to add a node at the beginning of a linked list
Know how to delete the first node in a linked list. It involves use of the delete command

Nodes in a Linked List

Be able to create a node that will be in a linked list
It should have a data part, and a link part

Traversing a Linked List

Be able to write this code *from memory*

```
    temp_node = linked_list;
    while (temp_node != NULL)
    {
        // Add code here to process the current node, which is temp_node
        temp_node = temp_node->next_node;
    }
```

Errors with Pointers

If you are given some program fragments with logic or syntax errors, be able to explain the errors
See page 327 for some examples