DASL 130 – C Programming Course

Lecture 4

Pointer Basics

- Declare a pointer: type *name;
- Note: position of star does not matter so long as it’s between type and name
- To point to a normal variable use ‘address of’ operator -- &
- int normal = 5;
- int * p = &normal;

Pointer Usage

- *p references the value at p
- p is the address stored in p, aka the address of the variable that p points to
- Be careful when using operators on pointers!
- *p++ is the value at the next address after p
- (*p)++ is the value p points to plus 1

Arrays

- Declaring an Array:
- type name[size]; //Values for each
- type name[] = {element1, element2, etc};
- type name[size] = {size must match #elements};

Array Access

- Arrays are zero-indexed – first element has the 0 subscript
- a[0] – first element of array ‘a[x]’
- a[4] – last element of a five element array ‘a[5]’
- for the array - int a[5] – the syntax a[6] will access memory out of bounds

Array Pointer Access

- int a[5] = {0, 1, 2, 3, 4};
- int *p = a (is the same as) int *p = &a[0]
- p++ is then &a[1]
- In other words, arrays are stored sequentially in memory

Strings

- char * a = “A string”
- char a[] = “A character array”
- char *a = new char a[x]
- Strings are fixed once declared, the contents of the string cannot be changed
- The pointer can be targeted to another string, however
- A char array can be changed