# Lecture 06 Notes

Derek Harter

CSci 515 Spring 2015 *<2015-02-18 Wed>*

## 1 First Session (11 - 11:40)

### 1.1 Introduction to Arrays

An array is a consecutive group of memory locations that all have the same type. When you declare an array, C will allocate a block of memory large enough to hold all of the values of the type you ask for. You access the elements of the array by using a subscript or index.

- Declaring an array

```
type arrayName[arraySize];
```

- Indexing an array
- Arrays in C are indexed **STARTING AT 0** (0 based indexing)
- The size of the array like `int c[10]` is 10 items, indexed from 0 to 9.
- When accessing all of the elements of an array, always use an indexed controlled for loop, and always index the loop from 0 up to N, like this:

```
int c[10];

for (int i = 0; i < 10; i++)
{
}
```

- It is good practice to declare a constant, and use it wherever you are referencing your array.

```
const int ARRAY_SIZE = 10;
int c[ARRAY_SIZE];

for (int i = 0; i < ARRAY_SIZE; i++)
{
}
```

- This way, if you need to change the size of the array and code processing the array elements, you change 1 single location, and all of the code will work correctly.
- And when I say good practice, read that to mean you SHOULD ALWAYS be doing this for assignments for this class.

# 2 Second Session (11:45 - 12:30)

## 2.1 Examples of Using Arrays

- Initializing with a loop
- Initialize to random values.
- Display all the values in an array.
- Finding the minimum of the values of an array
- Sum the values of an array

# 3 Third Session (12:40 - 1:40)

## 3.1 Passing Arrays to Functions

- Arrays are passed to function by reference (by default)
- **ALWAYS** pass the array and the size to function to process an array. Makes function self-contained, does not depend on any globals.

## 3.2 Advanced Array Processing

- Revisit our dice game example from a previous lab.
- Randomly initialize to range [1, 6]

- Example, pass (2) arrays as input and an array to hold the result.
- Count frequency of values.
- In statistics, combinations of uniformly occurring randomness cause normally distributed probabilities. This is an example (central limit theorem).
- A histogram is simply a bar chart of frequencies of occurring outcomes in some experiment. In our experiment, the possible outcomes are sums from 2 to 12.