**Chapter 3 - Array-Based Implementations - Answers**

Top of Form

Bottom of Form

**Content**

**Answers**

**1. When we pass an array to a function in C++, we normally use 2 parameters. That are those two parameters?**

We pass the array itself and the number of elements in the array. The parameter used to pass the array actually contains the memory address (location of) the first element of the array. For example, given the array:

    int list[5] = {1, 2, 3, 4, 5};

We might have a function called print that we call like this:

    print(list, 5);

The function might look like this:

    void print( int array[], int size)  
    {  
        for (int i=0; i<size; i++)  
            cout << array[i] << ' ';  
        cout << endl;  
    }

**2. When an array is not full of data (i.e., some array elements contain useful data and others do not), what is a common technique for keeping track of which parts of the array contain useful data and which parts are "empty"?**

We put the useful data at the beginning of the array (at the smaller indexes) and leave the unused/empty elements at the end of the array. We use a variable, say called size, to hold the number of elements that we are currently using. Array elements with indexes 0 through size - 1 contain useful data. Elements at indexes size through the end of the array are unused.