Learning Objectives

- Arrays holding multiple values
- Accessing array elements
- No bound checking in C++
- Array initialization
- Processing array contents
- Arrays as function arguments

Lab exercise 1

Unzip all codes from Week4 Example codes and observe the outcome of each code.

Lab Exercise 2

Please answer all the short questions below:

1. Define the following arrays: (only definition)
   a. empNums, a 100 element array of ints
   b. payRates, a 25 element array of floats
   c. miles, 14 element array of longs
   d. cityName, 26 element array of string objects

2. What is wrong with the following array definitions?

```
int readings[-1];
float measurements[4.5];
int size;
string names[size];
```

3. What would the valid subscript values be in a four-element array of double called s?
4. What is the difference between an array's size declarator and a subscript?
5. What is "array bounds checking"? Does C++ perform it?
6. What is the output of the following code?

```
int values[5], count;
for (count = 0; count < 5; count++)
    values[count] = count + 1;
for (count = 0; count < 5; count++)
    cout << values[count] << endl;
```

7. The following program skeleton contains a 20-element array of int s called fish. When completed, the program should ask how many fish were caught by fishermen 1 through 20, and store this data in the array. Complete the program.

```
#include <iostream>
using namespace std;
int main()
{
    const int NUM_FISH = 20;
    int fish[NUM_FISH];
    // You must finish this program. It should ask how
    // many fish were caught by fishermen 1-20, and
    // store this data in the array fish.
    return 0;
}
```

8. Given the following array definitions
   double array1[4] = {1.2, 3.2, 4.2, 5.2};double
   array2[4];
   will the following statement work? If not, why?
   array2 = array1;
9. When an array name is passed to a function, what is actually being passed?
10. When used as function arguments, are arrays passed by value?
11. The following program skeleton, when completed, will ask the user to enter 10 integers, which are stored in an array. The function avgArray, which you must write, is to calculate and return the average of the numbers entered.
   #include using namespace std;
   // Write your function prototype hereint main()
   { const int SIZE = 10;int
   userNums[SIZE];
     cout << "Enter 10 numbers: ";
   for (int count = 0; count < SIZE; count++)
   {
     cout << "#" << (count + 1) << " ";cin >>
   userNums[count];
    }
   cout << "The average of those numbers is ";cout <<
   avgArray(userNums, SIZE) << endl; return 0;
   }
   //
   //
   Write the function avgArray here.
    //