

Chapter 8 - Review of Arrays

Consider the following declarations:

```
int values[10];  
double scores[50];
```

Q1: How many elements are in the array values?

Q2: What is the range of index values for the array scores? Does everyone know what the question is asking?

Q3: How much memory is allocated for the array scores?

Q4: Write the C++ code that will initialize the array scores to 0.0

Arrays of Characters

Consider the following C++ code:

```
char alpha[26];

for (int i = 0; i < 26; ++i)
{
    alpha[i] = static_cast<char> ('A' + i);
}
```

Q1: Is this legal C++ code? If so, explain exactly what this code is doing. If not, why not?

Q2: The following C++ statement will compile correctly but will cause a problem when executed. What is the problem caused and how can this problem be corrected?

```
cout << alpha;
```

Parallel Arrays

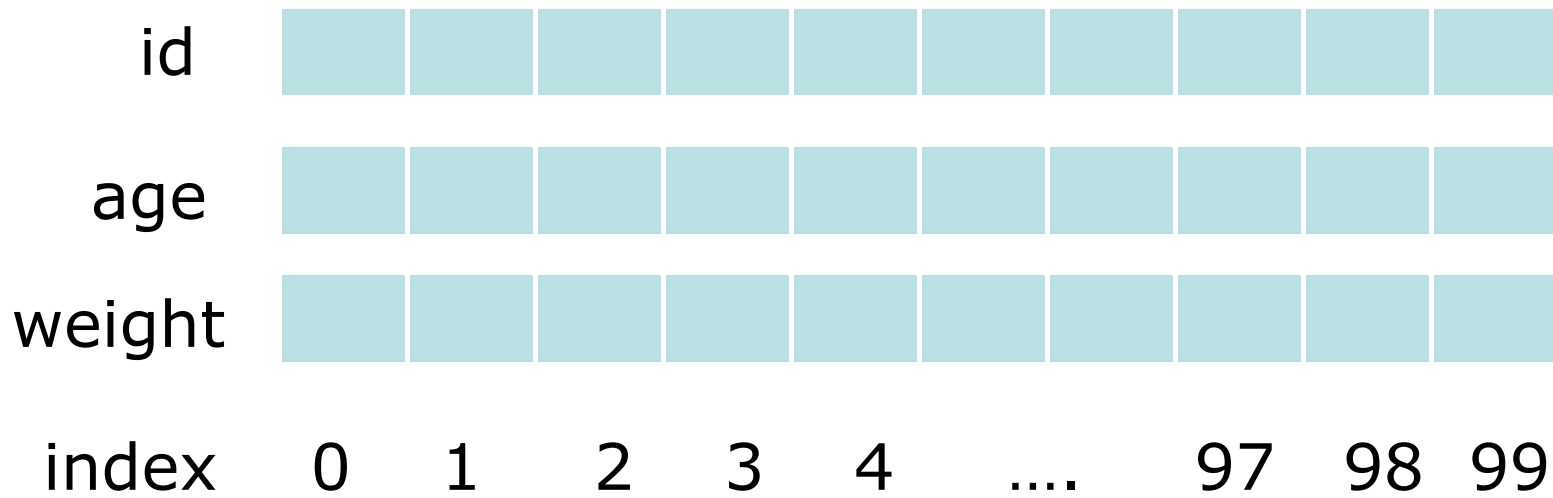
In CS150 we talked about the concept of parallel arrays. Remember, that parallel arrays are arrays of the same size that have related information in the same position within each array.

Let's assume that we wanted to be able to store the id #, age, and weight of 100 subjects. The C++ declaration would look like the following:

```
int id[100];  
int age[100];  
float weight[100];
```

Let's take a look at what these parallel arrays would look like.

Parallel Arrays Continued



Parallel Arrays, Functions, and Files

Problem: A datafile called athletes.txt exists which contains an unknown amount of information where each line of the file contains an id, age, and weight of a specific athlete. You are to write two functions as follows:

```
1) void readAthleteData (ifstream &inFile, int id[], int  
age[],  
float weight[], int &numAthletes);
```

This function reads in up to 100 lines of data into the previously mentioned parallel arrays and returns the number of athletes in the datafile.

```
2) int whatAge (const int age[], const int id[],  
int idNumber, int numAthletes);
```

This function returns the age of the athlete with the given idNumber.

Parallel Arrays, Functions, and Files Continued

```
#include <iostream>
#include <fstream>
#include <cstdlib>

using namespace std;

void readAthleteData (ifstream &inFile, int id[],
                    int age[], float weight[],
                    int &numAthletes);

int whatAge (const int age[], const int id[],
            int idNumber, int numAthletes);

const int MAX_ATHLETES = 100;
```

Parallel Arrays, Functions, and Files Continued

```
int main ()
{
    ifstream inFile;
    int id[MAX_ATHLETES], age[MAX_ATHLETES], numAthletes;
    float weight[MAX_ATHLETES];

    inFile.open ("athletes.txt");
    if (inFile.fail())
    {
        cout << "File Exception" << endl;
        exit (EXIT_FAILURE);
    }
    // Logic to read data & print an age goes here
    inFile.close ();
    return 0;
}
```

Parallel Arrays, Functions, and Files Continued

Let's take a look at `readAthleteData`.

```
void readAthleteData (ifstream &inFile,
                    int id[],
                    int age[],
                    float weight[],
                    int &numAthletes)
{
    numAthletes = 0;

    while ((numAthletes < MAX_ATHLETES) &&
           (inFile >> id[numAthletes]
            >> age[numAthletes]
            >> weight[numAthletes]))
    {
        ++numAthletes;
    }
}
```


Parallel Arrays, Functions, and Files Continued

Problem: Now you write the function whatAge

```
int whatAge (const int age[], const int id[],  
             int idNumber, int numAthletes);
```

This function returns the age of the athlete with the given idNumber number.