Array (std::array first)

***What is it, how they work?***

An array is a series of numbers/letters/symbols that are the same type (int, float, double, char etc…) that are held in a container in a strict linear sequence. And array does not hold any information other than the information that it is assigned. The way to create an array would look like:

char Scott [5]; // the best way to think of an array would be to think of it as a line of values

The char is the data type, Scottie is the name of the array (you can be as creative as you want with the name) and the [5] is the size of the array. So “char Scott [5];” can hold 5 pieces of data that are the same data type.

You can also hard code preset values inside the array. The way to create an array with preset values would look like:

char Scott [5] = {‘ S’, ‘c’, ‘o’, ‘t’, ‘t’};

int Scott [5] = {99, 5, 1, 22, 7};

The way array’s holds data is that it stars with 0 throw the assigned size -1(because you stared with 0). So in “char Scott [5]” the first slot would be Scott [0] and output the character S. And so on up to the last slot Scott [4], this outputs the last character t.

Matrix

***What is it, how they work?***

A matrix is the same thing as an array, but is an “array of arrays” (so looks like a box) matrices take arrays to a 2 dimensional field and more. The way matrices work is the same way as arrays but with more []’s. With a 3D matrix the first [ ] is referring to the row. The second [ ] is referring to the columns. The third [ ] would be the length. When you have any more[ ]’s there is no way we can think in the 4 dimensional to name what it is measuring, but you can still fill a 4D matrix, etc…

Matrices look like:

1. int Rich [3][3] // 2D
2. int Rich [10] [10] [10] // 3D
3. int Rich [10] [10] [10] [10] // 4D…etc

The way you fill a matrix is still on the same lines of arrays. You can hard code it or have the user input. The way you fill a matrix looks like:

From example 1:

int Rich [3][3]

hard coded by programmer://easiest

#include <iostream>

using namespace std;

int main()

{

int rich [3][3] = {{1,2,3},{4,5,6},{7,8,9}};// filling matrix with set numbers

cout <<rich[0][0]<<endl<<endl;// first number

cout <<" "<< rich[2][2];// last number

return 0;

}

This will see out the first and the last numbers in the matrix, which is 1 and 9. Because the spaces are still numbered 0-2.

filled by user:

#include <iostream>

#include<fstream>

using namespace std;

int jimmy(istream& giveme=cin)

{

int x;

cout << "fill me and hit enter after each int you enter";

giveme >> x;// inputting to matrix

return x;

}

int main()

{

int bob[3][3]; //creates a 3\*3 matrix

for(int i=0; i<3; i++) //This loops on the rows.

{

for(int j=0; j<3; j++) //This loops on the columns

{

bob[i][j] = jimmy();

}

}

for(int i=0; i<3; i++) //This loops on the rows.

{

for(int j=0; j<3; j++) //This loops on the columns

{

cout << bob[i][j] << " ";

}

cout << endl;

}

return 0;

}