C++ programming . . . continued:

using FUNCTIONS

```
Please enter X : 13

Please enter Y : 7

the sum of X and Y (X+Y) is:    20
the difference (X-Y) is:        6
the product (X*Y) is:           91
the quotient (X/Y) is:          1.85714

Press any key to continue
```

```cpp
// simple C++ program

#include <iostream>
using std::cin;
using std::cout;

int main() {

    double valueX, valueY;

    cout << "\n Please enter X : ";
    cin >> valueX;
    cout << "\n Please enter Y : ";
    cin >> valueY;

    cout << "\n the sum of X and Y (X+Y) is: \t " << valueX+valueY ;
    cout << "\n the difference (X-Y) is: \t" << valueX-valueY;
    cout << "\n the product (X*Y) is: \t\t" << valueX*valueY;
    cout << "\n the quotient (X/Y) is: \t" << valueX/valueY;
    cout << "\n\n";

    return 0;   }
```

```
Please enter X : 613

Please enter Y : 18

the sum of X and Y (X+Y) is:    631
the difference (X-Y) is:        595
the product (X*Y) is:           11034
the quotient (X/Y) is:          34.0556
the square root of X is :       24.7588

Press any key to continue
```

```
#include <iostream>
#include <cmath>
using std::cin;
using std::cout;

int main() {

    double valueX, valueY;

    cout << "\n Please enter X : ";
    cin >> valueX;
    cout << "\n Please enter Y : ";
    cin >> valueY;

    cout << "\n the sum of X and Y (X+Y) is: \t " << valueX+valueY ;
    cout << "\n the difference (X-Y) is: \t" << valueX-valueY;
    cout << "\n the product (X*Y) is: \t\t" << valueX*valueY;
    cout << "\n the quotient (X/Y) is: \t" << valueX/valueY;
    cout << "\n the square root of X is : \t" << sqrt(valueX);
    cout << "\n\n";

    return 0;   }
```

function call
(passing 1 parameter to it)

```
Please enter X : 6

Please enter Y : 3

the sum of X and Y (X+Y) is:    9
the difference (X-Y) is:        3
the product (X*Y) is:           18
the quotient (X/Y) is:          2
the square root of X is :       2.44949

Press any key to continue_
```

```
#include <iostream>
#include <cmath>
using std::cin;
using std::cout;

double howWeAdd(double,double);

int main() {

    double valueX, valueY;
    cout << "\n Please enter X : ";
    cin >> valueX;
    cout << "\n Please enter Y : ";
    cin >> valueY;
    cout << "\n the sum of X and Y (X+Y) is: \t " << howWeAdd(valueX,valueY) ;
    cout << "\n the difference (X-Y) is: \t" << valueX-valueY;
    cout << "\n the product (X*Y) is: \t\t" << valueX*valueY;
    cout << "\n the quotient (X/Y) is: \t" << valueX/valueY;
    cout << "\n the square root of X is : \t" << sqrt(valueX);
    cout << "\n\n";
    return 0;  }


double howWeAdd(double oneNum, double twoNum)
{
    return oneNum+twoNum;
}
```

this "**FUNCTION PROTOTYPE**" tells the compiler that *there will be a function* that the MAIN function will call, its name is howWeAdd, it takes two ARGUMENTS of type double, and it RETURNS A VALUE of type double

this is where we **CALL** THE FUNCTION, passing two PARAMETERS to it

this is the function

```cpp
// Double-subscripted array example.
#include <iostream>

using std::cout;
using std::endl;
using std::fixed;
using std::left;

#include <iomanip>

using std::setw;
using std::setprecision;

const int students = 3;    // number of students
const int exams = 4;       // number of exams

// function prototypes
int minimum( int [][ exams ], int, int );
int maximum( int [][ exams ], int, int );
double average( int [], int );
void printArray( int [][ exams ], int, int );

int main()
{
   // initialize student grades for three students (rows)
   int studentGrades[ students ][ exams ] =
      { { 77, 68, 86, 73 },
        { 96, 87, 89, 78 },
        { 70, 90, 86, 81 } };

   // output array studentGrades
   cout << "The array is:\n";
   printArray( studentGrades, students, exams );

   // determine smallest and largest grade values
   cout << "\n\nLowest grade: "
        << minimum( studentGrades, students, exams )
        << "\nHighest grade: "
        << maximum( studentGrades, students, exams ) << '\n';

   cout << fixed << setprecision( 2 );

   // calculate average grade for each student
   for ( int person = 0; person < students; person++ )
      cout << "The average grade for student " << person
           << " is "
           << average( studentGrades[ person ], exams )
           << endl;

   return 0;  // indicates successful termination

} // end main
```

```cpp
// find minimum grade
int minimum( int grades[][ exams ], int pupils, int tests )
{
   int lowGrade = 100; // initialize to highest possible grade

   for ( int i = 0; i < pupils; i++ )

      for ( int j = 0; j < tests; j++ )

         if ( grades[ i ][ j ] < lowGrade )
            lowGrade = grades[ i ][ j ];

   return lowGrade;
} // end function minimum
```

```cpp
// find maximum grade
int maximum( int grades[][ exams ], int pupils, int tests )
{
   int highGrade = 0;  // initialize to lowest possible grade

   for ( int i = 0; i < pupils; i++ )

      for ( int j = 0; j < tests; j++ )

         if ( grades[ i ][ j ] > highGrade )
            highGrade = grades[ i ][ j ];

   return highGrade;
} // end function maximum
```

```cpp
// determine average grade for particular student
double average( int setOfGrades[], int tests )
{
   int total = 0;
   // total all grades for one student
   for ( int i = 0; i < tests; i++ )
      total += setOfGrades[ i ];
   return static_cast< double >( total ) / tests;  // average
} // end function maximum
```

```cpp
// Print the array
void printArray( int grades[][ exams ], int pupils, int tests )
{
   // set left justification and output column heads
   cout << left << "                [0]  [1]  [2]  [3]";
   // output grades in tabular format
   for ( int i = 0; i < pupils; i++ ) {
      // output label for row
      cout << "\nstudentGrades[" << i << "] ";
      // output one grades for one student
      for ( int j = 0; j < tests; j++ )
         cout << setw( 5 ) << grades[ i ][ j ];
   } // end outer for
} // end function printArray
```

```
The array is:
                 [0]  [1]  [2]  [3]
studentGrades[0] 77   68   86   73
studentGrades[1] 96   87   89   78
studentGrades[2] 70   90   86   81

Lowest grade: 68
Highest grade: 96
The average grade for student 0 is 76.00
The average grade for student 1 is 87.50
The average grade for student 2 is 81.75
Press any key to continue
```