```
// SquareRoot.cpp - CALCULATE A SQUARE ROOT BY TRIAL AND ERROR
//
// MODULE INDEX
// NAME             CONTENTS
// SquareRoot       Calculate the square root of a number by trial and error
// main             Testing main line
//
// MAINTENANCE HISTORY
// DATE         PROGRAMMER AND DETAILS
// 06-09-11 JS  Original
//
//-----------------------------------------------------------------------------

#include <cstring>      // C-style string manipulation functions
#include <cstdlib>      // C-style standard library
#include <iostream>     // C++ input/output stream declarations
using namespace std;        // Expand the standard namespace
#include "dtime.h"      // Double time functions

//-----------------------------------------------------------------------------

// CALCULATE THE SQUARE ROOT OF A NUMBER BY TRIAL AND ERROR

double
SquareRoot (
    double      x)      // Value to be square rooted
{
    double      l, u;       // Lower and upper limits
    double      m;      // Midpoint
    double      m2;     // Midpoint squared

    if (x < 0) {
        cerr << "Error: square root of negative number\n";
        abort ();
    }
    if (x == 0)
        return 0;
    if (x < 1.0) {
        l = x;
        u = 1.0;
    } else if (x > 1.0) {
        l = 1.0;
        u = x;
    } else
        return 1.0;
    while (fabs((l - u) / l) >= 1.0e-8) {
        m = (l + u) / 2.0;
        m2 = m * m;
        if (m2 > x)
            u = m;
        else
            l = m;
    }
    return (l + u) / 2.0;
}

//-----------------------------------------------------------------------------

// TESTING MAIN LINE
```

```
int
main ()
{
    double       x;       // The value to be square rooted

    for (x = 0.0; x < 20.0; x += 0.5)
        cout << "x=" << x << "  root=" << SquareRoot(x) << '\n';
    return 0;
}
```