```
// Program #2 - Newton's method for solving f(x)=0

#include <iostream>
#include <cstdlib>
#include <cmath>
using namespace std;

double f( double x )
{
    return( 10.0*x*x + 2.0*x - 3.0 );
}
double fp( double x )
{
    return( 20.0*x + 2.0 );
}

int main()
{
// User iput: x1 = Initial guess
    double x1 = 5.0;
// ******************************
    const double TOLERANCE = 1.0e-10;
    const double REL_ERR_THRESH = 10.0;
    const int MAX_ITERS = 50;

    double fpx1, x2, error;
    char err_flag;
    int count = 0;
    cout.precision( 10 );

    do {
        fpx1 = fp( x1 );
        if ( fpx1 == 0.0 ) {
            cout << "\nSTOP: Zero derivative encountered at x = " << x1 << "\n";
            break;
        }
        x2 = x1 - f( x1 ) / fpx1;
        count++;
        if ( abs( x2 ) > REL_ERR_THRESH ) {
            error = abs( ( x2 - x1 ) / x2 );
            err_flag = 'R';
        }
        else {
            error = abs( x2 - x1 );
            err_flag = 'A';
        }
        x1 = x2;
        cout << "x" << count << " = " << x2
             << "   Error type = " << err_flag
             << "   Approx error = " << error << endl;
        if ( count > MAX_ITERS ) {
            cout << "\nSTOP: Maximum number of iterations exceeded.\n";
            break;
        }
    } while ( error > TOLERANCE );

    cout << "\n\n"; system( "PAUSE" );
    return( EXIT_SUCCESS );
}
```