Embedded computing for scientific and industrial imaging applications

Lecture 4 - C demo, Newton’s method

Outline

C demo with visual studio

- Computing square roots by Newton's method
- Incremental developing by using git

Computing square roots

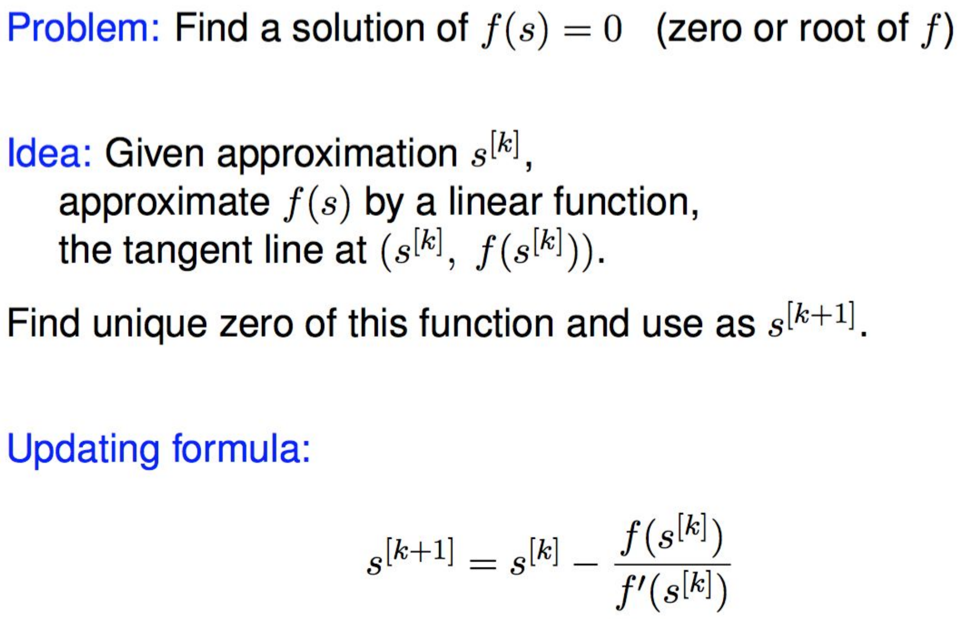
Hardware arithmetic units can add, subtract, multiply, divide. Other mathematical functions usually take some software.

$\sqrt{2} \approx 1.4142135623730951$

In most languages, sqrt(2) computes this.

```
#include <math.h>

sqrt(2);
```

Newton’s method



Problem: Find a solution of $f(s) = 0$ (zero or root of f)

Idea: Given approximation $s^{[k]}$,
approximate $f(s)$ by a linear function,
the tangent line at $(s^{[k]},\ f(s^{[k]}))$.

Find unique zero of this function and use as $s^{[k+1]}$.

Updating formula:

$$s^{[k+1]} = s^{[k]} - \frac{f(s^{[k]})}{f'(s^{[k]})}$$

Approximate s = √x

Newton's method to find root of $s^2 - x = 0$.

```
double s = 1.;
for (int i = 0; i < maxiter; i++)
{
    s = 0.5 * (s + x / s);
}
```

where `maxiter` is some maximum number of iterations.

Demo

Goals:

- Develop our own version of sqrt function.
- Start simple and add complexity in stages.
- Illustrate some C programming.
- Illustrate use of git to track our development
- Get familiar with visual studio IDE

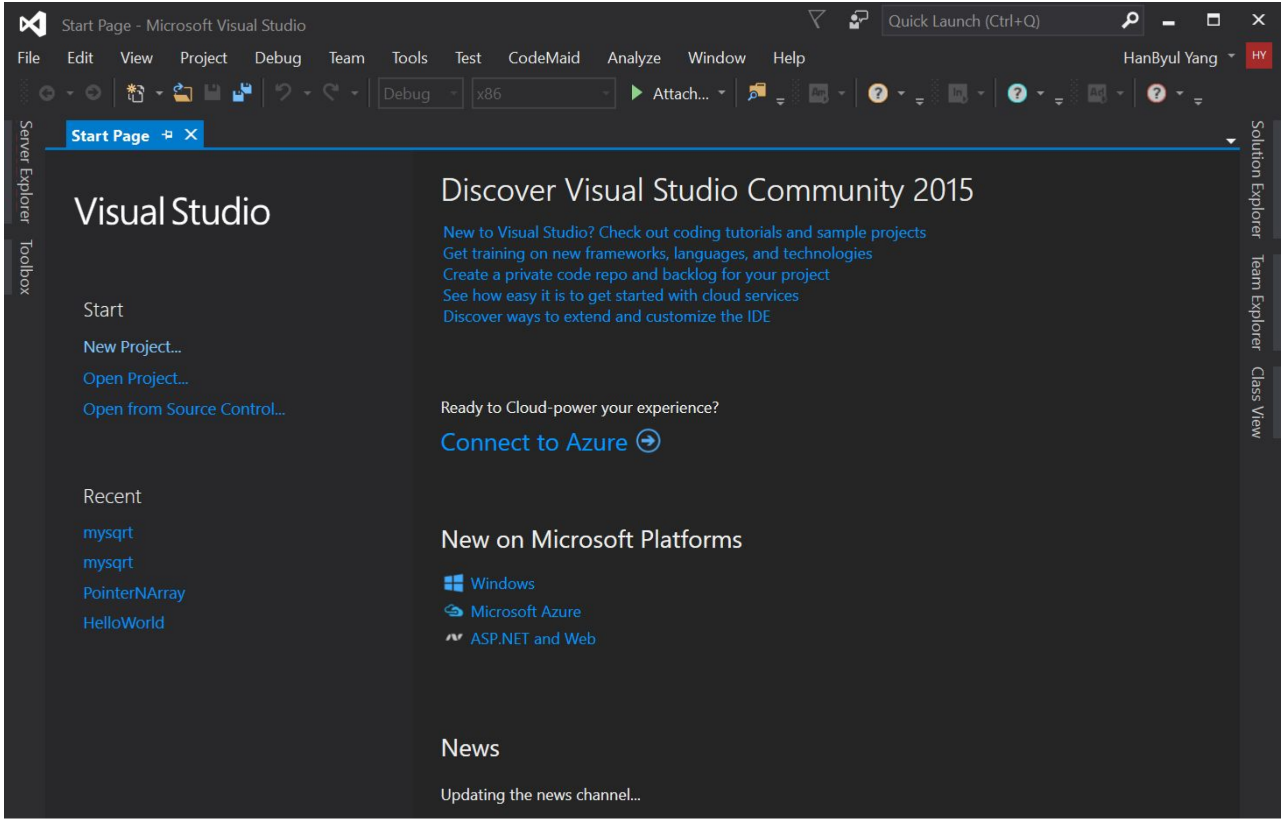
We will do this in codes/mysqrt directory so you can examine the various versions later.

Demo : mysqrt.c

```c
#include <stdio.h>
#include <math.h>

int main()
{
    double x = 2;
    double s = sqrt(x);
    printf(“sqrt(%f) = %f\n”, x, s);
}
```

Demo : visual studio

Start “New Project”



Visual Studio

Start

New Project...

Open Project...

Open from Source Control...

Recent

mysqrt

mysqrt

PointerNArray

HelloWorld

Discover Visual Studio Community 2015

New to Visual Studio? Check out coding tutorials and sample projects

Get training on new frameworks, languages, and technologies

Create a private code repo and backlog for your project

See how easy it is to get started with cloud services

Discover ways to extend and customize the IDE

Ready to Cloud-power your experience?

Connect to Azure

New on Microsoft Platforms

Windows

Microsoft Azure

ASP.NET and Web

News

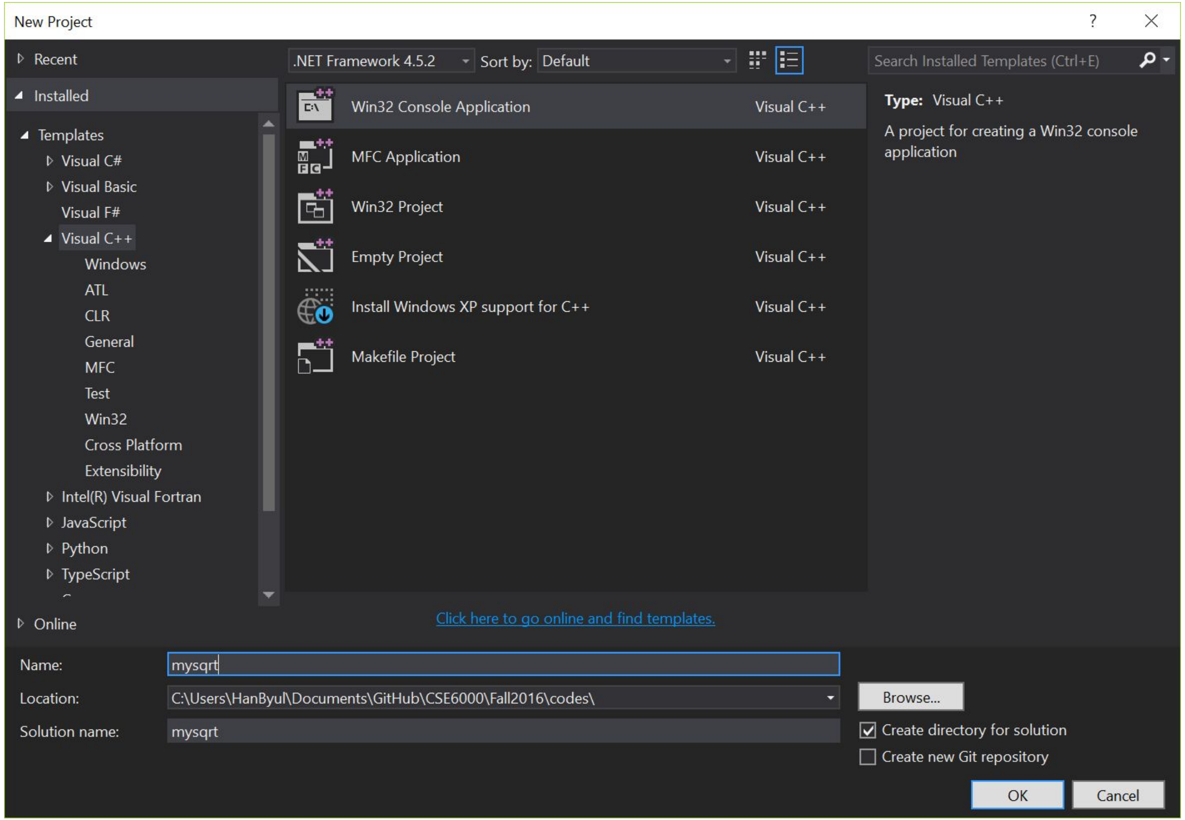
Updating the news channel...

Demo : new project

Visual c++

Win32 Console application

“mysqrt”



Type: Visual C++

A project for creating a Win32 console application

Name: mysqrt

Location: C:\Users\HanByul\Documents\GitHub\CSE6000\Fall2016\codes\

Solution name: mysqrt

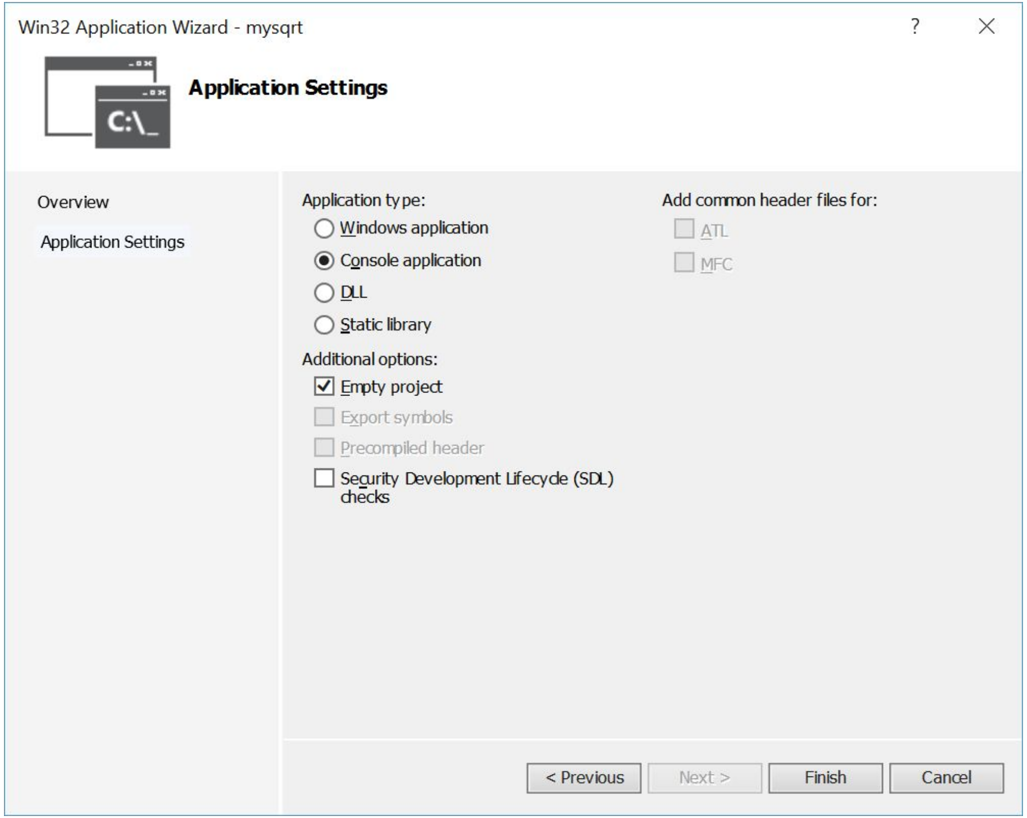
☑ Create directory for solution

☐ Create new Git repository

Demo : application setting

Console application

Empty project



Win32 Application Wizard - mysqrt

Application Settings

Overview

Application Settings

Application type:

- ○ Windows application
- ● Console application
- ○ DLL
- ○ Static library

Additional options:

- ☑ Empty project
- □ Export symbols
- □ Precompiled header
- □ Security Development Lifecycle (SDL) checks

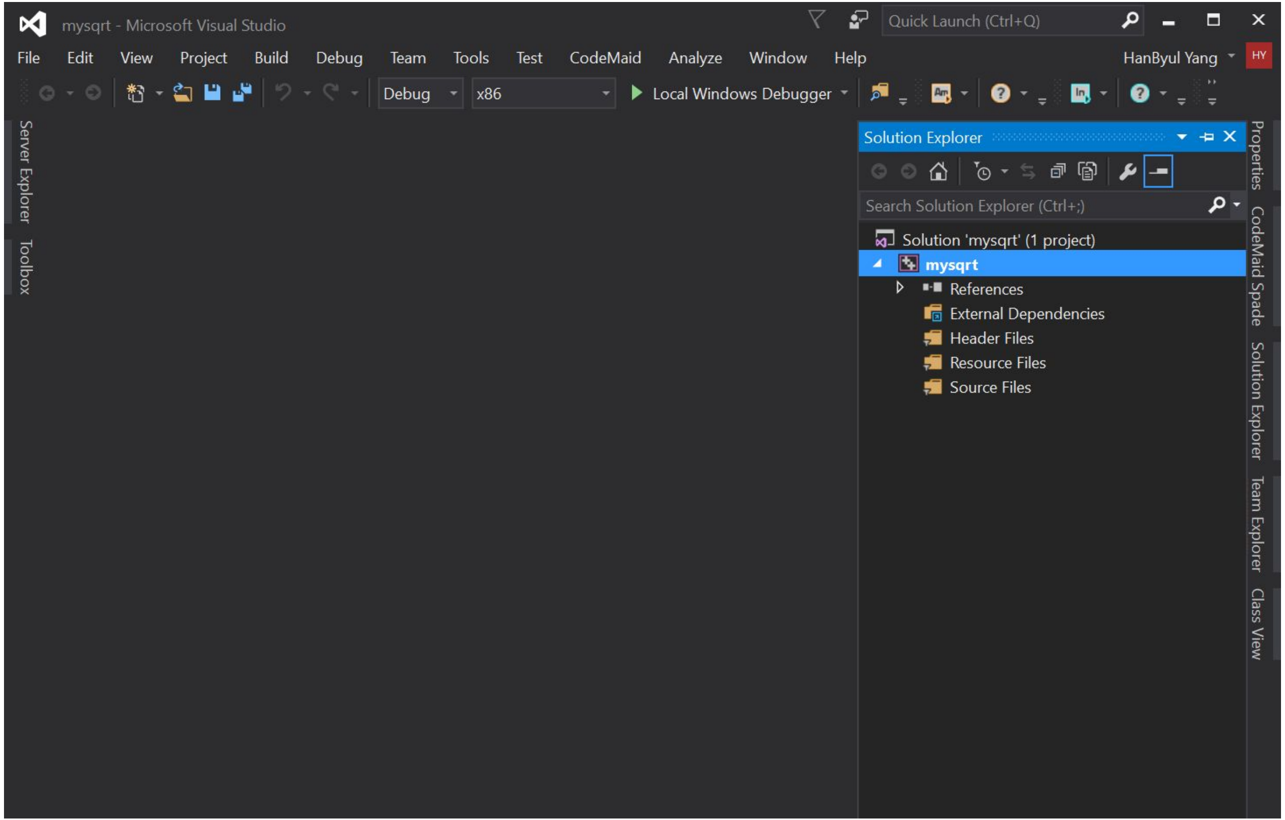
Add common header files for:

- □ ATL
- □ MFC

Demo : Solution Explorer

Solution

Project

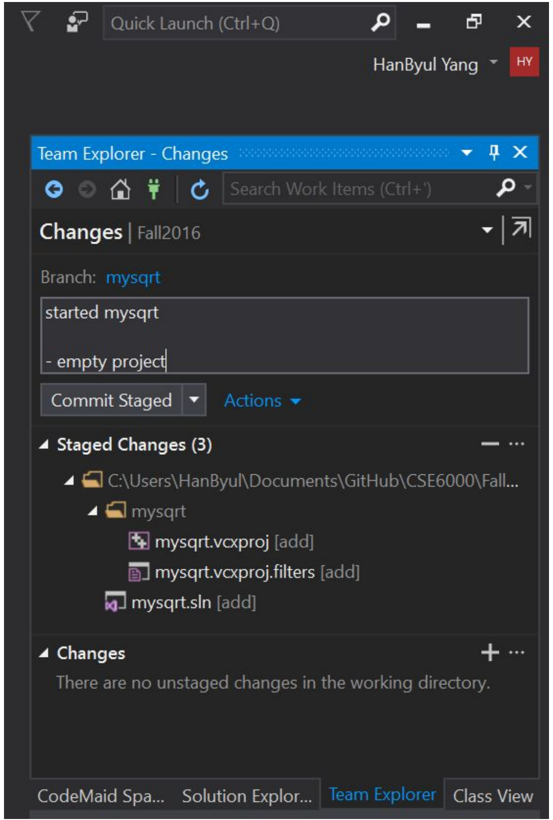


Demo : Team Explorer

Branch : mysqrt

Commit 3 files

- mysqrt.sln
- mysqrt.vcxproj
- mysqrt.vcxproj.filters



Branch: mysqrt

```
started mysqrt

- empty project
```

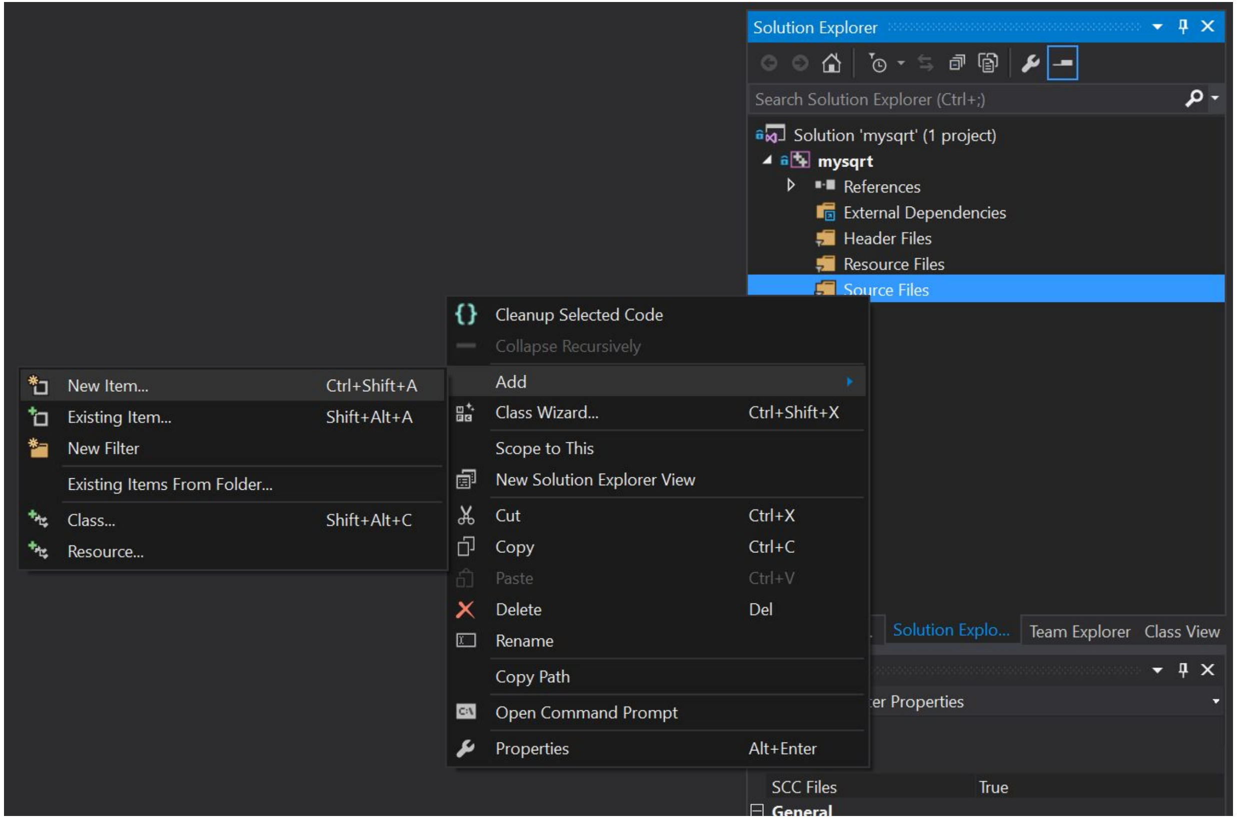
Staged Changes (3)

```
C:\Users\HanByul\Documents\GitHub\CSE6000\Fall...
  mysqrt
    mysqrt.vcxproj [add]
    mysqrt.vcxproj.filters [add]
  mysqrt.sln [add]
```

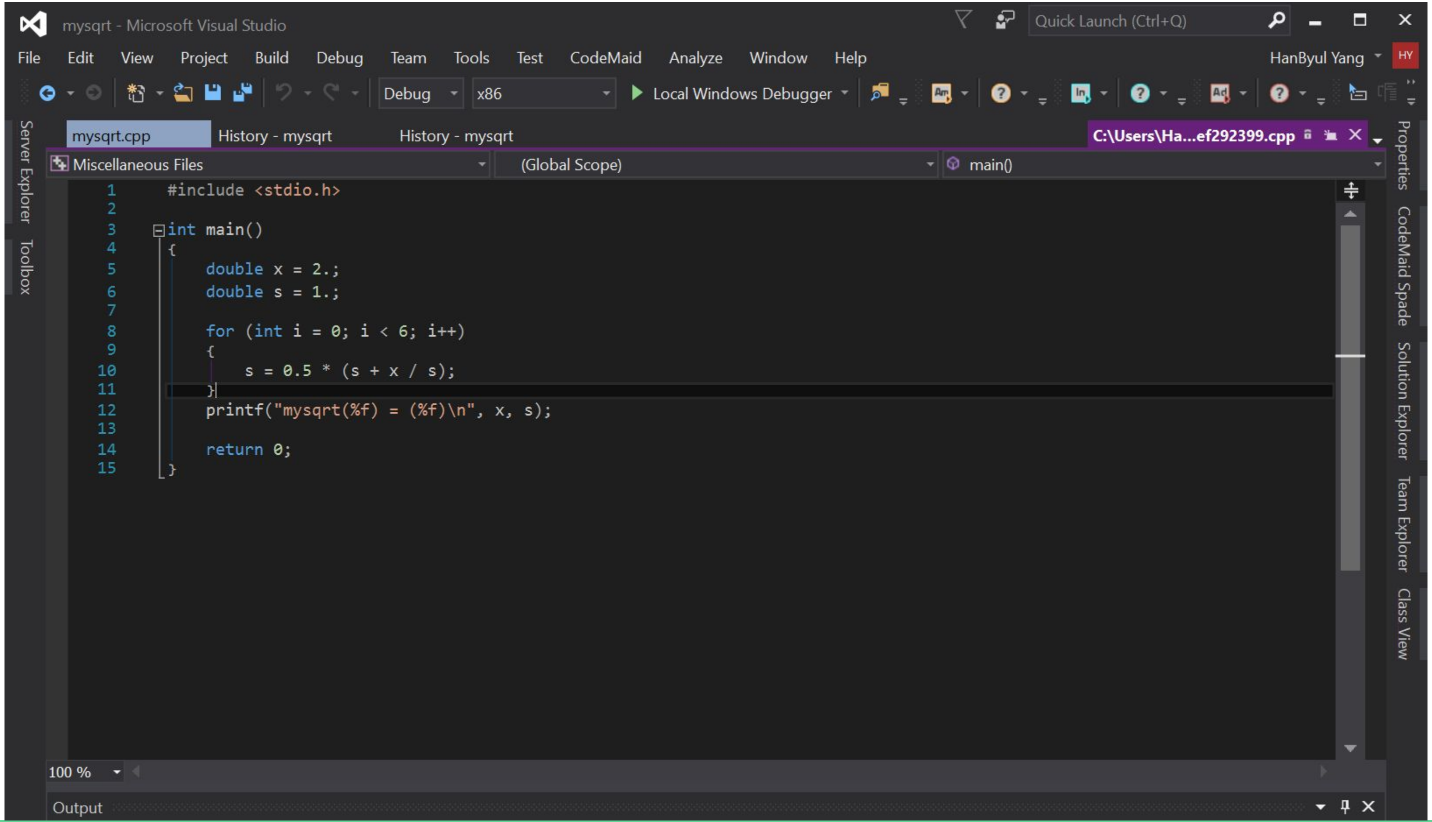
Changes

There are no unstaged changes in the working directory.

Add “New Item”

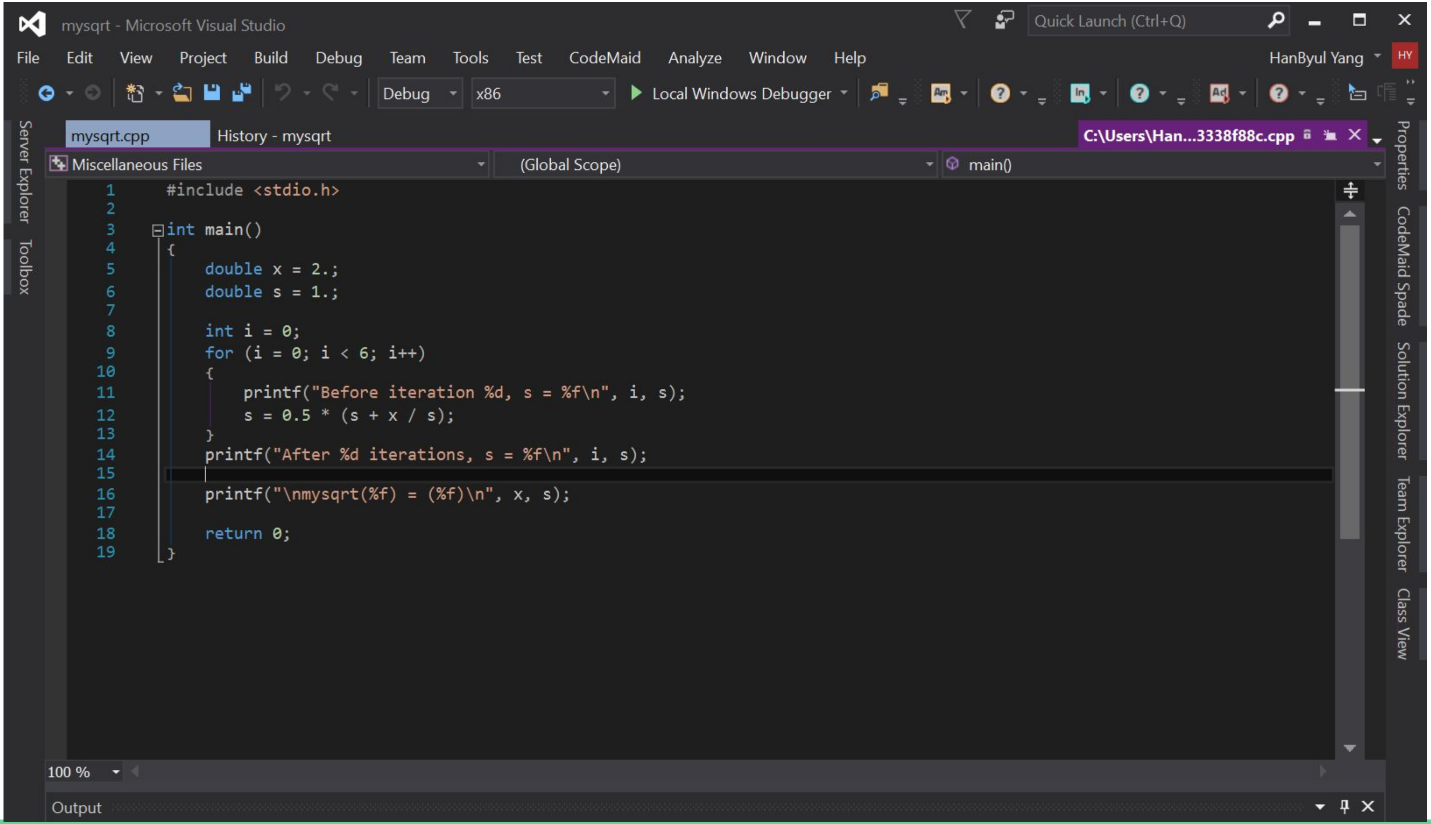

Solution Explorer
Search Solution Explorer (Ctrl+;)
Solution 'mysqrt' (1 project)
mysqrt
References
External Dependencies
Header Files
Resource Files
Source Files
Cleanup Selected Code
Collapse Recursively
Add
Class Wizard... Ctrl+Shift+X
Scope to This
New Solution Explorer View
Cut Ctrl+X
Copy Ctrl+C
Paste Ctrl+V
Delete Del
Rename
Copy Path
Open Command Prompt
Properties Alt+Enter
New Item... Ctrl+Shift+A
Existing Item... Shift+Alt+A
New Filter
Existing Items From Folder...
Class... Shift+Alt+C
Resource...
Solution Explo...
Team Explorer
Class View
er Properties
SCC Files
True
General

Demo : First version of mysqrt



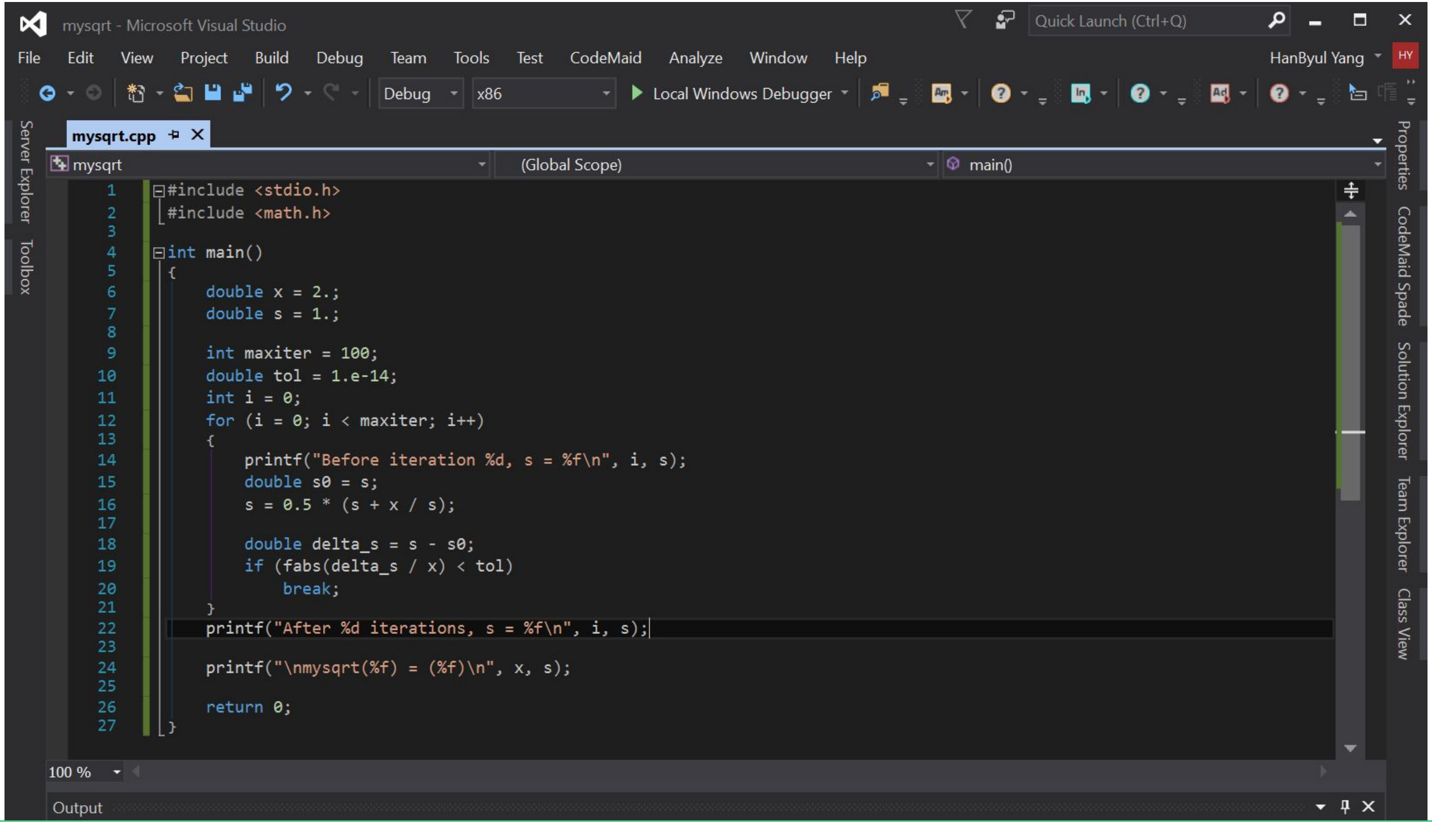
```
#include <stdio.h>

int main()
{
    double x = 2.;
    double s = 1.;

    for (int i = 0; i < 6; i++)
    {
        s = 0.5 * (s + x / s);
    }
    printf("mysqrt(%f) = (%f)\n", x, s);

    return 0;
}
```

Demo : Print each iteration



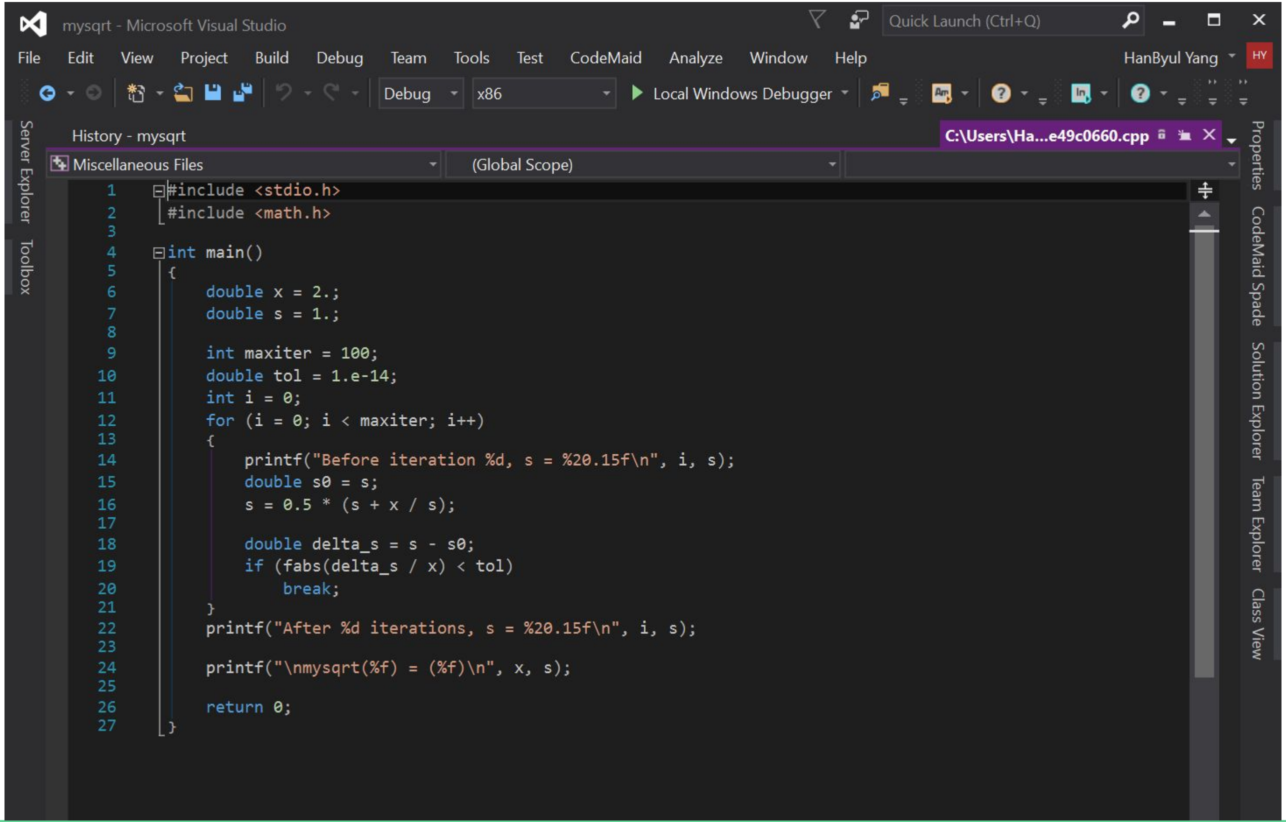
```
#include <stdio.h>

int main()
{
    double x = 2.;
    double s = 1.;

    int i = 0;
    for (i = 0; i < 6; i++)
    {
        printf("Before iteration %d, s = %f\n", i, s);
        s = 0.5 * (s + x / s);
    }
    printf("After %d iterations, s = %f\n", i, s);

    printf("\nmysqrt(%f) = (%f)\n", x, s);

    return 0;
}
```

Demo : Add convergence test



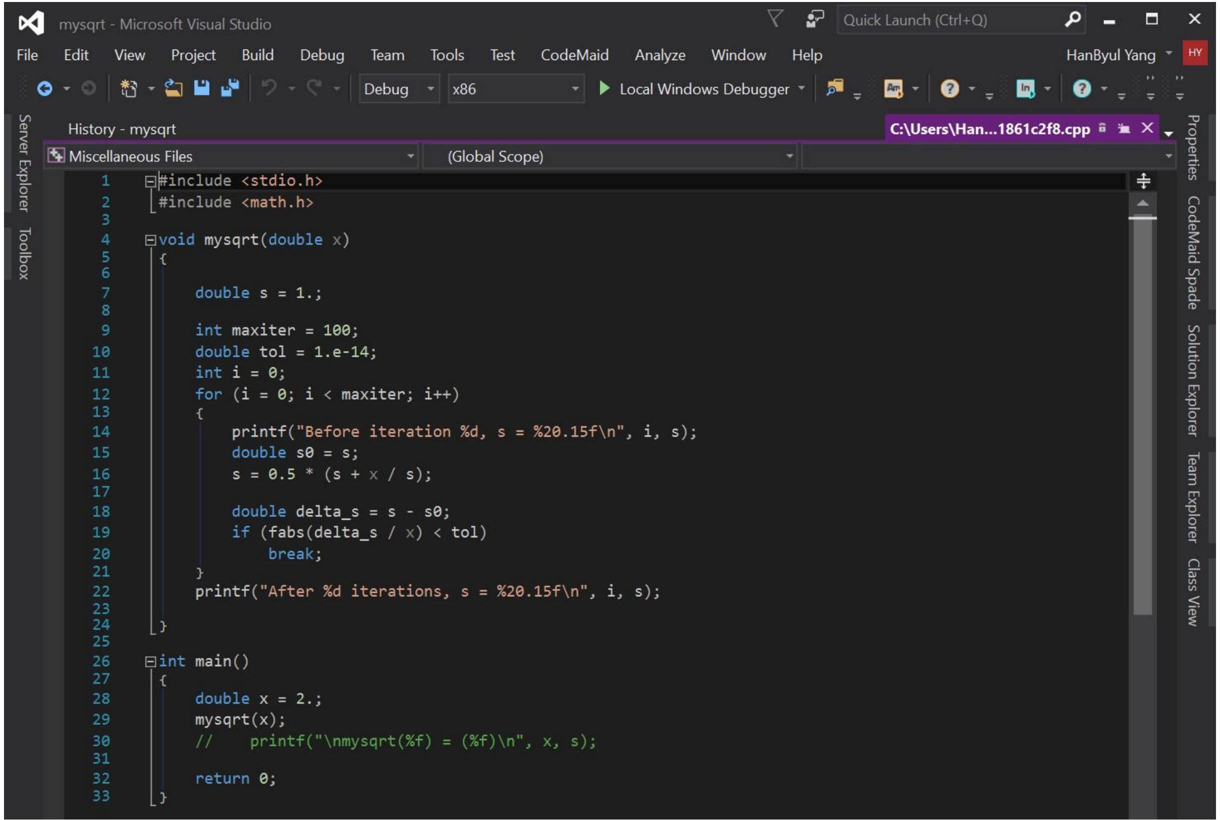
```
#include <stdio.h>
#include <math.h>

int main()
{
    double x = 2.;
    double s = 1.;

    int maxiter = 100;
    double tol = 1.e-14;
    int i = 0;
    for (i = 0; i < maxiter; i++)
    {
        printf("Before iteration %d, s = %f\n", i, s);
        double s0 = s;
        s = 0.5 * (s + x / s);

        double delta_s = s - s0;
        if (fabs(delta_s / x) < tol)
            break;
    }
    printf("After %d iterations, s = %f\n", i, s);

    printf("\nmysqrt(%f) = (%f)\n", x, s);

    return 0;
}
```

Demo : Refinement of printing floating point number



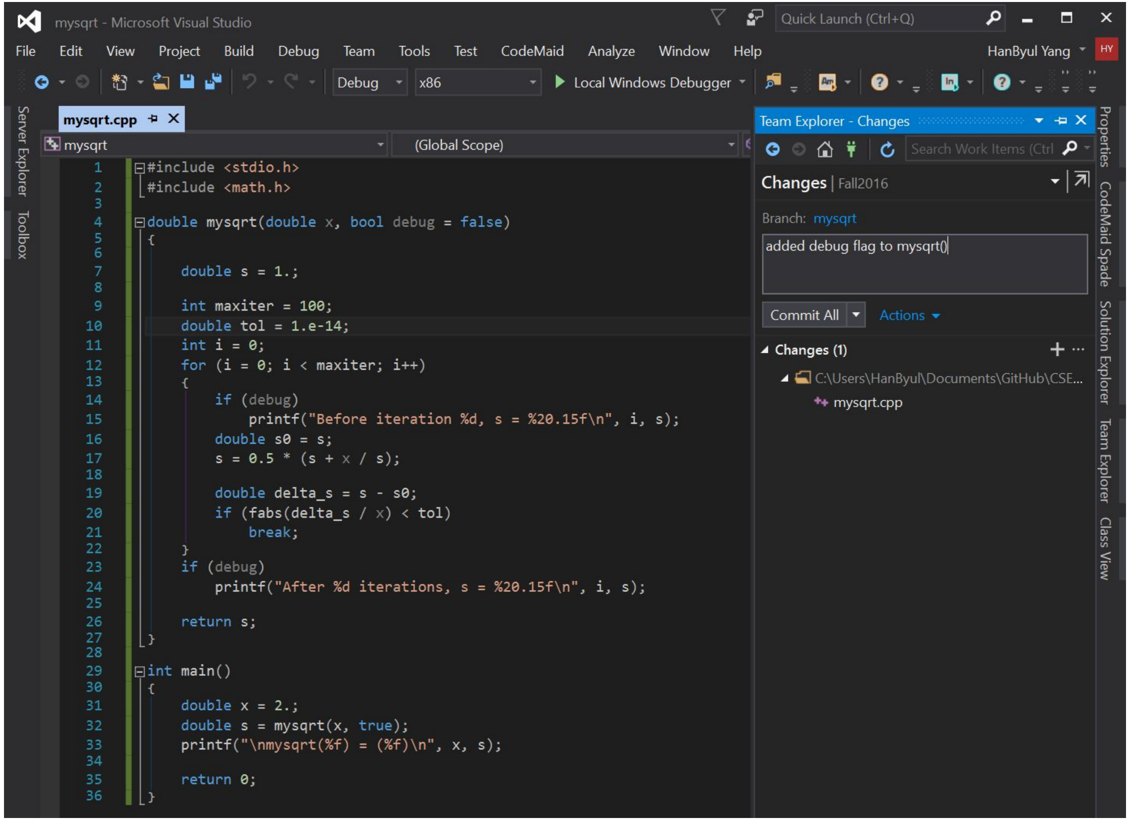
```c
#include <stdio.h>
#include <math.h>

int main()
{
    double x = 2.;
    double s = 1.;

    int maxiter = 100;
    double tol = 1.e-14;
    int i = 0;
    for (i = 0; i < maxiter; i++)
    {
        printf("Before iteration %d, s = %20.15f\n", i, s);
        double s0 = s;
        s = 0.5 * (s + x / s);

        double delta_s = s - s0;
        if (fabs(delta_s / x) < tol)
            break;
    }
    printf("After %d iterations, s = %20.15f\n", i, s);

    printf("\nmysqrt(%f) = (%f)\n", x, s);

    return 0;
}
```

Demo : mysqrt() function



```
#include <stdio.h>
#include <math.h>

void mysqrt(double x)
{

    double s = 1.;

    int maxiter = 100;
    double tol = 1.e-14;
    int i = 0;
    for (i = 0; i < maxiter; i++)
    {
        printf("Before iteration %d, s = %20.15f\n", i, s);
        double s0 = s;
        s = 0.5 * (s + x / s);

        double delta_s = s - s0;
        if (fabs(delta_s / x) < tol)
            break;
    }
    printf("After %d iterations, s = %20.15f\n", i, s);

}

int main()
{
    double x = 2.;
    mysqrt(x);
    //    printf("\nmysqrt(%f) = (%f)\n", x, s);

    return 0;
}
```

Demo : mysqrt() function with debug flag



```
#include <stdio.h>
#include <math.h>

double mysqrt(double x, bool debug = false)
{

    double s = 1.;

    int maxiter = 100;
    double tol = 1.e-14;
    int i = 0;
    for (i = 0; i < maxiter; i++)
    {
        if (debug)
            printf("Before iteration %d, s = %20.15f\n", i, s);
        double s0 = s;
        s = 0.5 * (s + x / s);

        double delta_s = s - s0;
        if (fabs(delta_s / x) < tol)
            break;
    }
    if (debug)
        printf("After %d iterations, s = %20.15f\n", i, s);

    return s;
}

int main()
{
    double x = 2.;
    double s = mysqrt(x, true);
    printf("\nmysqrt(%f) = (%f)\n", x, s);

    return 0;
}
```

Changes | Fall2016

Branch: mysqrt

added debug flag to mysqrt()

Changes (1)

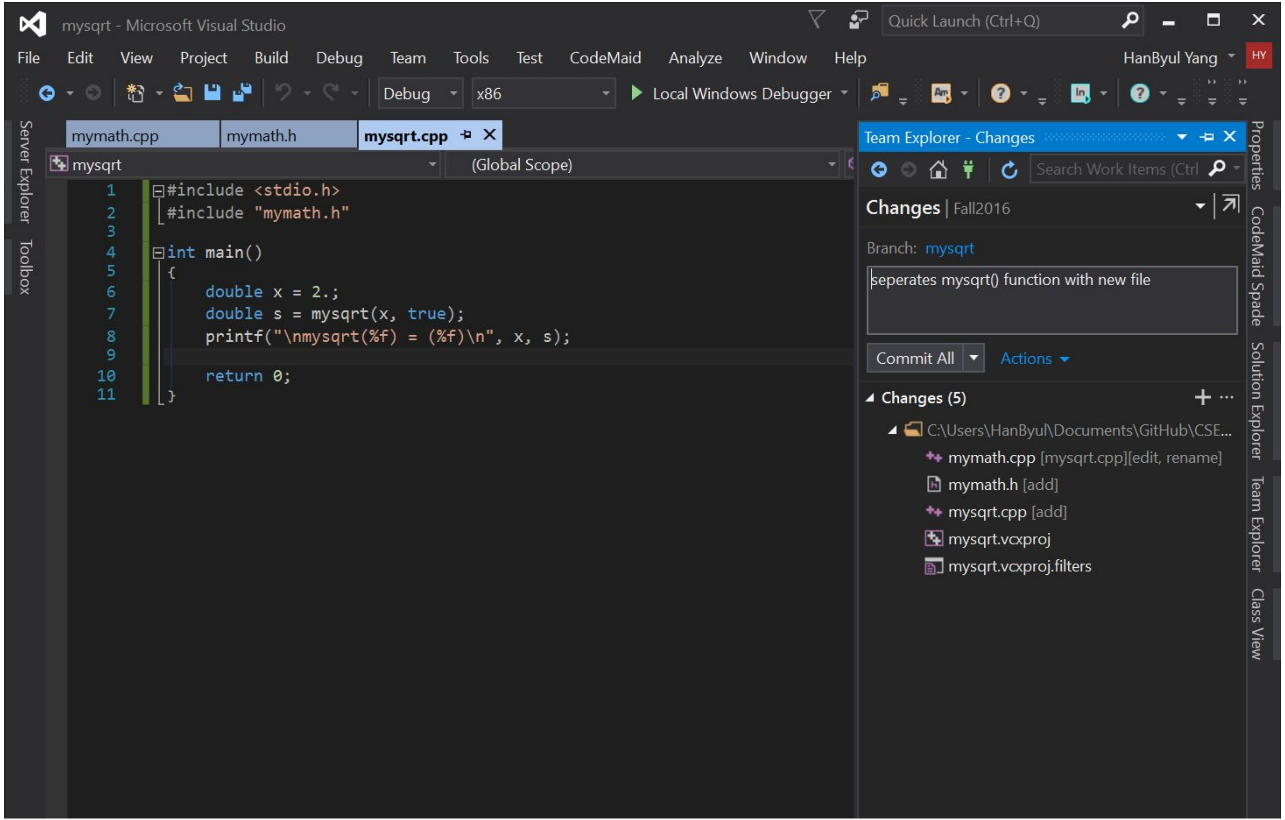
C:\Users\HanByul\Documents\GitHub\CSE...

mysqrt.cpp

Demo : separates mysqrt()

New files

- mymath.h
- mymath.cpp



```
#include <stdio.h>
#include "mymath.h"

int main()
{
    double x = 2.;
    double s = mysqrt(x, true);
    printf("\nmysqrt(%f) = (%f)\n", x, s);

    return 0;
}
```

Changes | Fall2016

Branch: mysqrt

seperates mysqrt() function with new file

Changes (5)

C:\Users\HanByul\Documents\GitHub\CSE...

- mymath.cpp [mysqrt.cpp][edit, rename]
- mymath.h [add]
- mysqrt.cpp [add]
- mysqrt.vcxproj
- mysqrt.vcxproj.filters

Links

- Getting Started with C++ in Visual Studio
- Visual C++ in Visual Studio 2015
- C++ Language Reference