

# CS 150 Lab 3

## Arithmetic and the Debugger

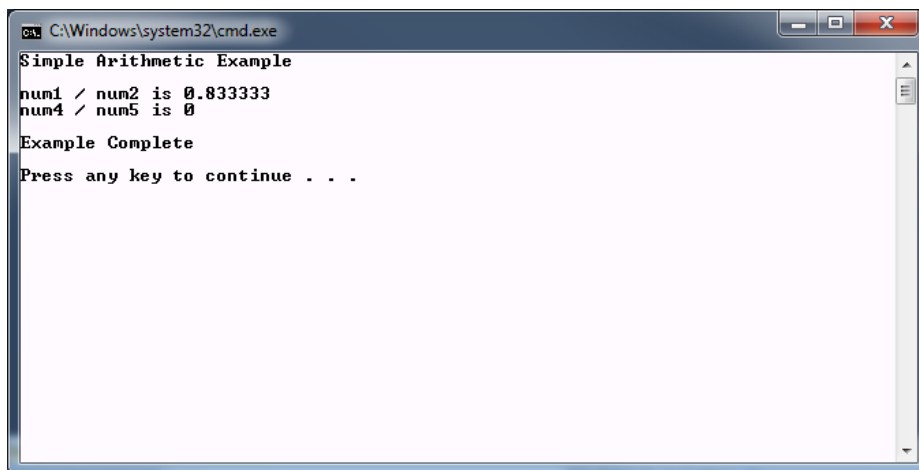
The main objective of today's lab is to use some basic mathematics to solve a few real world problems. In doing so, you are to begin getting accustomed to using the Visual Studio debugger.

- Be sure your output looks exactly like the specified output.
- Be sure to submit the completed project to **CS150-01 Lab** when you are done.
- Be sure to use the program skeleton and add comments to your code!
- Show the instructor or TA your solution to each problem before submitting.
- You do not need to submit the challenge question!

### Lab 3.0

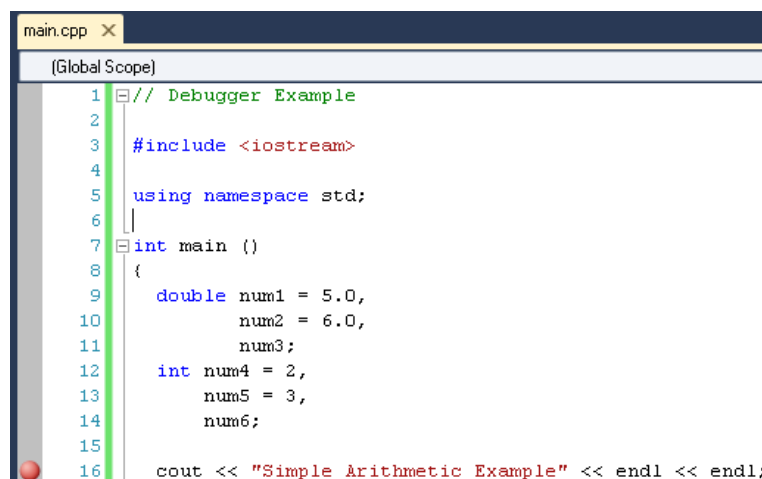
We are going to begin using the Visual Studio 2010 debugger to aid with debugging programs.

1. Go to **CS150-01 Public** folder and drag a copy of the solution **DebugEx** to your desktop.
2. Build your solution and then execute the program using the command "Start Without Debugging." You will see the following.



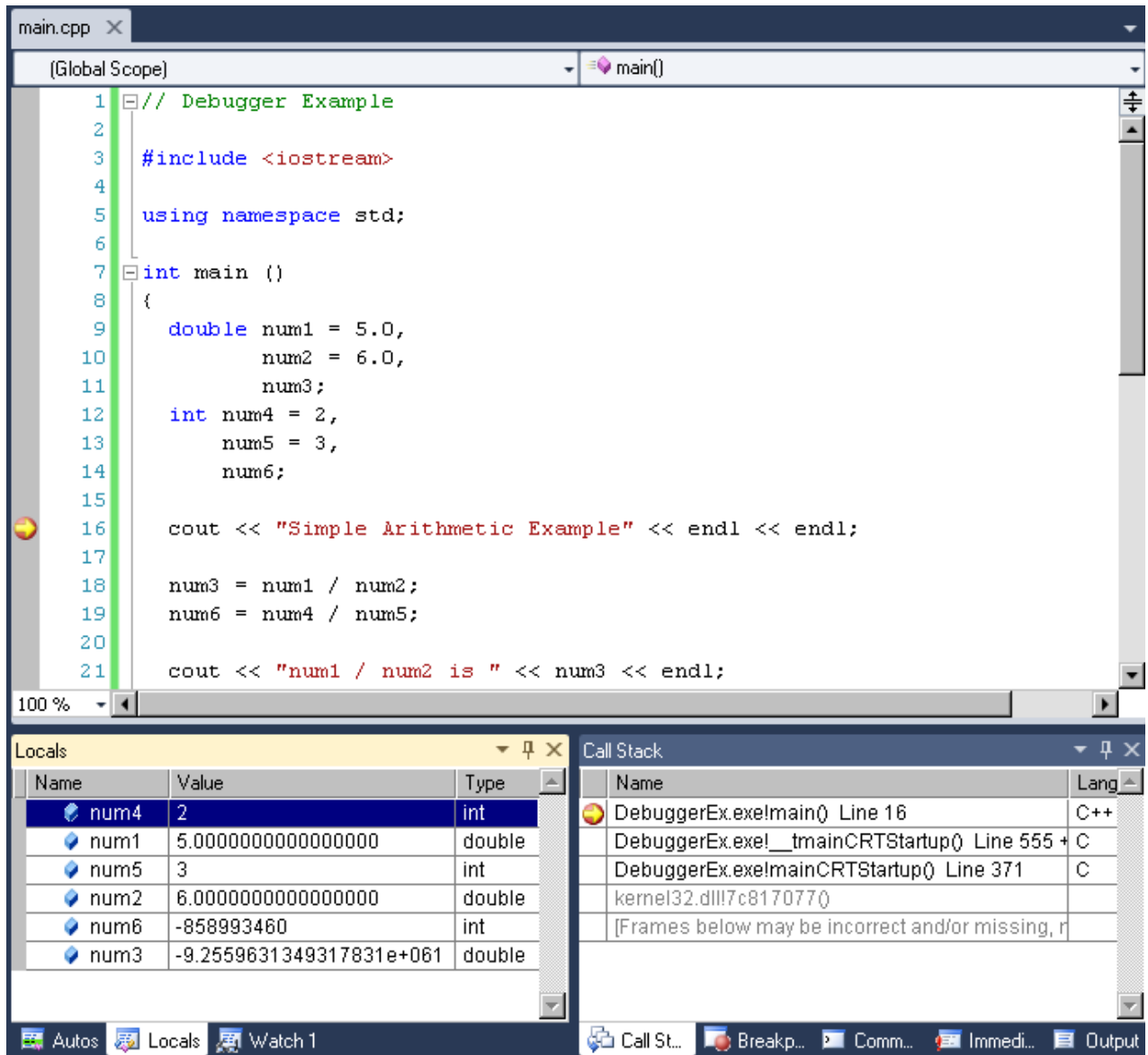
```
C:\Windows\system32\cmd.exe
Simple Arithmetic Example
num1 / num2 is 0.833333
num4 / num5 is 0
Example Complete
Press any key to continue . . .
```

3. In looking at your output you wonder why num4 / num5 is 0 since both numbers are nonzero. What is going on? Let's debug!
4. Close the output window and go back to the code. Set a breakpoint by clicking an executable statement in the light blue column as follows. Notice the red button next to the executable statement cout on line 16.



```
main.cpp X
(Global Scope)
1 // Debugger Example
2
3 #include <iostream>
4
5 using namespace std;
6
7 int main ()
8 {
9     double num1 = 5.0,
10         num2 = 6.0,
11         num3;
12     int num4 = 2,
13         num5 = 3,
14         num6;
15
16     cout << "Simple Arithmetic Example" << endl << endl;
```

- Let's start the debugger and step through our program one executable statement at a time. Start the debugger by going to Debug → Start Debugging. The debugger will stop at the first executable statement that has a breakpoint. Your screen should look something like the following. Make sure the Locals tab is selected and not the Autos tab.



**6. Show your instructor/TA the above screen.**

- Let's talk about what we are looking at.
- From here we can continue running the program until the next breakpoint by selecting Resume as shown below.



- You can also just execute the next statement by selecting Step Over.
- We will talk about Step Into and Step Out when we get to functions.
- Execute each statement up to but not including the return statement.
- STOP – Show the instructor or TA**

## Lab 3.1: Fahrenheit to Celsius

Add a project called **03\_1\_TemperatureConversion** to your **PUNetIDLabs** solution. This project is to convert a Fahrenheit temperature to both Celsius and Kelvin. The conversion formulas are:

$$C = \frac{5}{9}(F - 32)$$

$$K = \frac{5}{9}(F - 32) + 273$$

Sample Input/Output where the Fahrenheit temperature is the only input shown in bold below.

```
*****  
Temperature Converter  
*****  
  
Enter a non-negative Fahrenheit temperature: 212  
  
212 Fahrenheit is 100 Celsius  
  
212 Fahrenheit is 373 Kelvin
```

1) Briefly describe the data that your program will need to use.

---

---

2) List the variable declarations necessary to store the data listed in 1.

---

---

3) Open up Visual Studio and write a complete program for the above problem.

4) Show the instructor/TA that you can step through your solution with the debugger.

**► STOP – Show the instructor or TA**

## Lab 3.2: Time Converter

Add a project called **03\_2\_TimeConverter** to your **PUNetIDLabs** solution. This project is to convert a number of seconds into hours, minutes, and seconds in a format like HH:MM:SS.

Sample Input / Output where the total number of seconds is the only input shown in bold below.

```
*****  
Time Converter  
*****  
  
Enter seconds since day began: 100  
  
The time is 0:1:40
```

1) Briefly describe the data that your program will need to use.

---

---

---

2) List the variable declarations necessary to store the data listed in 1.

---

---

---

3) Open Visual Studio and write a complete program for the above problem.

**► STOP – Show the instructor or TA**

### Optional Challenge:

Always print your time as HH:MM:SS so the above time would show 00:01:40

Once both projects are completed, place your solution **PUNetIDLabs** into the **CS150-01 Drop** folder on Turing. Your solution is to have all six (2 from lab 1, 2 from lab 2, 2 from lab 3) projects completely working and correct.