
CSI33 Data Structures

Sharon Persinger

Fall 2019

Day 20 November 13



Topics – Classes in C++

Operator overloading

Class variables

Class methods

Assignment 5: Complete the implementation of Rational Class

Setting up a project in Visual Studio with the Rational Class

This process requires that the C++ add-ins are installed.

First, create a Windows Console App. Choose Create a New Project , Select Visual C++, then Windows Console Application. Name your project, and store it in a location that you control

Make sure the Solution explorer window is visible. Turn it on from the View Menu if necessary

Next add a class. From the Project menu, choose Add Class. Enter the class name Rational into the dialog. The files Rational.h and Rational.cpp will be created and placed into the appropriate folders in the Solution. The files will have tabs in the editing window.

Copy the code from Rational.h on the webpage into the Rational.h file. Copy the code from Rational.cpp into the Rational.cpp file. Copy the Code from RationalNum.cpp into the source file for the console application. Save all of this.

Overloading operators

Operators can be overloaded as a method in the class. See Rationalv2. `operator+` is overloaded as a member method of the Rational class.

Rational v2

Operators can be overloaded as stand-alone methods. Some operators must be overloaded this way, for instance, `input >>` and `output <<` operators.

Rational v3

See Rationalv3. `operator+` is overloaded as a stand-alone function. `operator>>`, `operator<<` are declared in the class as friend, prototypes not in class. These operators must be overloaded as stand-alone functions.

Class variables and methods

Class variables are declared using the keyword `static` in the class definition with the other variables. There is one copy for the class of the member variable, shared with all of the object instances of the class.

See `Card.h`, `Card.cpp`

Class variables are declared outside of any functions, are initialized once at first execution of the program. Notice in this example the static variables are also declared `const`. They are also `private`, so accessible only through the public member methods of the class.

A class can have a static method also. Static methods can access class variables, but not instance variables.

Assignment 5: Complete the Rational class.
