Integer math q2

# Revision history

V 1.0: initial release

# usage history

# problem description

The following code was submitted by a student who claimed that integer division was avoided. The student is incorrect. What went wrong in his reasoning? Determine two test cases that can be used to prove that integer division is still encountered. Fix the code so that integer division is indeed avoided.

#include <iostream>

**using** **namespace** std**;**

int main**()**

**{**

int a**,** b**;**

double result**;**

cout **<<** "Enter values for a and b!" **<<** endl**;**

cin **>>** a **>>** b**;**

result **=** a**/**b**;**

cout **<<** result**;**

**}**

# Solution

The student incorrectly assumed that because the variable “result” was a double, the computer would case **the calculation of the quotient** to a double. In reality, the computer will first evaluate the right hand side of the expression, in which both variables are integers. This will introduce integer division errors. The result will get cast to a double only after the computation of the right hand side is complete.

The student can test the code with many test cases. Two suggestions are:  
  
x = 1, y = 12  
x = 10, y = 3

This code can be modified in several ways to avoid the integer division. Ultimately, the result must be cast to a double before it the computation is carried out. One potential method to solve the problem is shown below:

#include <iostream>

**using** **namespace** std**;**

int main**()**

**{**

int a**,** b**;**

double result**;**

cout **<<** "Enter values for a and b!" **<<** endl**;**

cin **>>** a **>>** b**;**

result **=** 1.0\*a**/**b**; //Note that I am implicitly casting here**

cout **<<** result**;**

**}**

# suggested test cases

* none

# required topics

* Integer division
* Designing good test cases
* Casting rules