COSC 1030

C++ operators/arithmetic
(adapted from Dr. Kim Buckner)

Nels Frazier

University of Wyoming

Aug. 31, 2022

Labs and Assignments.

- Do not submit comments unless you REALLY need to.
- They may or may not get read.
- Meaningful comments are OK, but they should provide clarity and value to the **semantics** of your program.
- DO FOLLOW INSTRUCTIONS!!!!!!!
- Ask for help if you need it.

Cut and Paste

- As some will find, this does not always work.
- If you try to copy the code from the PDF document you may have gotten errors.
- This is NOT a bug, it is a feature.
- Short answer, just be careful.

Basic arithmetic

- This should be (and is) straight forward.
- We write computations in a program similar to how we might write a formula.
- Assignment is right-to-left however.
- It does NOT mean equality except in the most broad sense.

Objectives today

- Examine the rules for operator precedence of the simple arithmetic operators of C++.
- Examine the use of parentheses for altering the default precedence ordering for arithmetic operator computations.
- Note the facts of integer division in C++.

Operator precedence

- The order of operation for the standard binary arithmetic operators *****, **/**, **%**, **+**, **-** is given by the precedence table shown in Appendix B of the text.
- What does this mean to you?

CompSurprise.cpp

```
// CompSurprise.cpp
// Kim Buckner
// COSC 1030
// Lecture 02
// Operator precedence, integer division.
#include<iostream>
using std::cout;
using std::endl;
int main()
{
  cout << "Do these computations make sense?" << endl;
  cout << "2+7/4 is " << 2+7/4 << endl;
  cout << "(2+7)/4 is " << (2+7)/4 << endl;
  cout << "8-9/5 is " << 8-9/5 << endl;
  cout << "8+(-9/5) is " << 8+(-9/5) << endl;
  return 0;
}
```

Comments

- Because division has precedence over addition, and because integer division returns an integer, note the program output.
- Integer division is a gotcha that **will** occur.

Computing an Average

Objectives

- Examine sentinel controlled repetition design with a **while** control structure.
- Reason with program design in a simple domain.

The problem

- Display average of an arbitrarily long sequence of integers.
- User is inputting data from the keyboard.
- The **sentinel value** (indicating finish) is **–9999**.
- The sentinel value is NOT part of the computation.

Thinking about the problem

Pseudocode

- Set a counter and an accumulator (temporary sum) to zero

Thinking about the problem

Pseudocode

- Set a counter and an accumulator (temporary sum) to zero
- Prompt the user to enter the first integer
- While the integer which is input is not the sentinel value of –9999,
  - add the integer to the accumulator,
  - increment the counter,
  - prompt for and 'get' another integer

(more . . .)

- *If* the value of the counter is greater than zero,
  - compute average by dividing the accumulator by the counter value,

(more . . .)

- *If* the value of the counter is greater than zero,
  - compute average by dividing the accumulator by the counter value,
  - display the average.
- *else*
  - prompt user to input at least one integer before the sentinel value
  - or . . .

§CompAvg.cpp, BadLogic.cpp

What you should do.

- Read/refer to the document on the home page of the WyoCourses site titled "Pseudocode Guide."
- Resources currently available at www.cs.uwyo.edu/∼nfrazie1/cosc1030/
- Get the code for *CompSurprise.cpp*.
- Play with it.

(more . . .)

- Try other combinations of arithmetic operators, integers and parentheses.
- Get your hands dirty.
- Program 01 instructions are posted.
- This program is due by midnight Friday.