Laboratory III –Data type conversions, expressions, and mathematical functions

Larry Caretto
Computer Science 106

Computing in Engineering and Science

February 14, 2006

California State University Northridge

Outline

- Review exercise two
- Exercise three goals
- Summarize information on data types
- Review lecture material on expressions, operator precedence and conversion
- Outline tasks for exercise three

Review Exercise Two

- Note that division by zero produces infinity expressed as 1.#INF
- Zero divided by zero is indefinite 1.#IND
- Integer division truncates
- Cannot enter decimal or E notation numbers as input for integers
- Learned how to produce spacing in output

Multiple Data Inputs

- Can use space or enter key between inputs
- Can have one or more input commands
  - cin >> x >> y;
  - cin >> x; cin >> y;
- For either the single or repeated cin commands above you can use a space or enter between data inputs

Exercise Three Goals

- Understand how to handle type conversion in expressions
- Learn how to write expressions to represent equations correctly
- Be able to define and use symbolic constants
- Be able to use mathematical functions from the C++ library

Expressions

- ***<variable>*** = ***<expression>***;
- Expression is a constant, a variable, or a collection of variables, constants and operators
- Mathematical operators, in order of precedence are [1] unary – (highest), [2] multiplication (*), division (/), and mod (%), [3] addition (+) and subtraction(-)

Expressions (continued)

- Use parentheses to override normal operator precedence
- Can use extra parentheses for clarity
- Example equation

$$w = \frac{u+v}{x+y}$$

- Code is `w = ( u + v ) / ( x + y );`
- Without parentheses we would get incorrect result

California State University Northridge 7

Converting Data Types

- Conversion occurs when a value of one type is assigned to another type
  - int x = 6; double y = x; // result is y = 6.0
  - double u = 7.89; v = u; // result is v = 7
- When two operands are different types, the lower type is promoted to the higher type. (*E. g*. <*double*> <*operator*> <*int*> gives a <*double*> result.)

California State University Northridge 8

Converting data types

- Expressions evaluated with no knowledge of left-hand side
- Can force conversion by functions like int() and double().
  - double z = 1/2; // gives z = 0.0
  - double y = double(1)/2; // gives z = 0.5
- Watch out for expressions like KE = (1/2) * m * V * V;

California State University Northridge 9

Tasks for Exercise Three

- One – copy and paste code with various conversions; correct one error; run and study results
- Two – Write a program to get results of three expressions for three sets of data
- Three and four – write programs using math functions and a symbolic constant for π to get relationships for a circle.

California State University Northridge 10

Task One – Copy and Paste

```
double a; int x = 27, y = 4, z;
z = x / y;
cout << "Example of output from different"
     << " data types. \nIn these results "
     << "x = 27, y = 4, and z are type "
     << " int; \na is type double.\n\n";
cout << "For z = x / y, z = " << z;
a = x / y;
cout << "\nFor a = x / y, a = " << a;
z = double( x ) / y;
cout << "\nFor z = double( x )/ y, z = "
```

Study and understand the results

California State University Northridge 11

Task Two

- Write a program that
  - Inputs w, x, y, and z
  - Computes a, b, and c
  - Prints the results
- Execute for three data sets
- Compute results with calculator and make sure that your program is correct

$$a = \frac{w+x}{y-z}$$

$$b = \frac{z}{x}\frac{w-x}{2y}$$

$$c = \frac{w-\frac{x}{y}}{\frac{z}{x}+wy}$$

California State University Northridge 12

Tasks Three and Four

- Three – compute circumference, C, and area, A, from an input radius, r
  - C = $2\pi r$ and A = πr^2
- Four – compute radius from input area
- For both tasks use symbolic constant `PI = 4 * atan(1.0)`
- Use `pow` and `sqrt` functions
  - Examples of C++ math functions

California State University Northridge 13

Mathematical Functions

- Use `#include <cmath>` for library
- Compute power x^y using `pow(x,y)`
  - Note order: pow(number,power)
    - `pow(4,3)` gives $4^3 = 64$
    - `pow(3,4)` gives $3^4 = 81$
- Other functions include
  - square root [`sqrt(x)`]
  - e^x [`exp(x)`] and natural log [`log(x)`]
  - absolute value [`fabs(x)`]
    - Note use of fabs!!! abs(x) gives type int result

California State University Northridge 14

Trigonometric Functions

- In most programming languages (including C++) arguments to trig functions are expressed in radians
  - There are 2π radians in circle (360^o)
  - π radians = 180^o
  - $\pi/4$ radians = 45^o
- Common trig functions
  - Sine, cosine and tangent are `sin(x)`, `cos(x)` and `tan(x)`
  - Arctangent, $\tan^{-1}$, is `atan(x)`

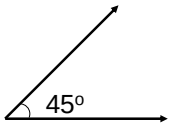
California State University Northridge 15

Symbolic Constants

- Define type and use keyword `const`
  - `const int MAX_VALUES = 10;`
  - `const double PI = 4 * atan( 1.0 );`
- `const` by itself gives an integer constant which may be an error (but is not a syntax error caught by the compiler)
  - `const PI = 4 * atan( 1 );` gives PI = 3
- Convention: all caps for symbolic const
- See text for additional discussion

California State University Northridge 16

Why is $\pi = 4\tan^{-1}(1)$?



- The tangent of a 45^o angle = 1
- $45^o = \pi/4$ radians (used in C++)
- $\tan(\pi/4) = 1$ (in radians)
- $\tan^{-1}(1) = \pi/4$
- $\pi = 4\tan^{-1}(1)$

California State University Northridge 17