Exercise 2 – Numerical data types and simple input/ output

Larry Caretto
Computer Science 106

Computing in Engineering and Science

February 7, 2006

California State University Northridge

Outline

- Review exercise one (can hand in this week as late assignment)
- Present assessment results
- Review lecture topics for exercise two
  - Basic structure of simple C++ programs
  - Basic input and output
- Instructions for preparing file to submit
- List tasks for exercise two

Review of Exercise One

- You should have learned how to use the Visual C++ IDE to enter, compile, link, and execute programs
- You saw a sample C++ program with
  - A semicolon to end each statement
  - Braces {} to separate blocks of code
  - Use of cin >> and cout << for keyboard input and screen output

Review Syntax Errors

- Click on error message to see line
- Error messages are not always clear
- Error may occur on line above line identified by error message
- Single error may give multiple error messages
- Learn how to interpret error messages during the semester

Assessment Results

- 10 students completed assessment
  - 8 rate general computer skills above minimal
  - 8 students rate programming skills as minimal or none
- Two students do not have Math 150A or higher
- All students have access to a computer at home (6 windows PC, 2 MAC 2 No answer or answer unclear)

Assessment Quiz

- $\int x^3 dx = x^4/4 + C$ – Three incorrect (three more missing constant, C)
- $d(e^{ax})/dx = ae^{ax}$ – Four correct
- $100011_2 = 35_{10}$ – One correct; use binary numbers to understand storage
- x = 10; x = x + 5; gives x = 15 – Six correct; shows meaning of = as replacement of value in computing

Assessment Quiz

- File definition – Four partially complete
  - a collection of information, which may be programs, data or images, stored on permanent storage (e. g., a hard drive, solid-state USB device, zip drive, CD or floppy) that can be accessed by a unique name.
- Rounding 123.4567 to four significant figures produces 123.5 or 1.235×10^2 – Everyone correct!

California State University Northridge 7

Exercise Two Goals

- Understand differences between different data types (double and int)
- Learn results of division by zero
- Learn effect of entering a fraction as an input for an int-data-type variable
- Be able to write input and output
  - Multiple input values with one cin
  - Output statements with spaces and new lines

California State University Northridge 8

Basic C++ Program Structure

```
#include <iostream>
using namespace std;
int main()
{
  <your program statements>
  return EXIT_SUCCESS;
}
```

- Case sensitive; ignores spaces and new lines

California State University Northridge 9

Numerical data types

- Binary representation of numbers
- Floating point data types (float, double and long double)
  - Have decimal points and wide range
  - Division behaves in expected manner
- Integer data types (int, short int, long int, unsigned int)
  - Have narrow range
  - Integer division truncates

California State University Northridge 10

Review Output using cout

- `cout << "<string>";` writes the string between the quotation marks to the screen
- `cout << x;` writes the value of the variable, x, to the screen.
- Can have one or more output (<<) operators in a single `cout` command

Code	*Screen output*
cout << "Name"	Name
int x = 2; cout << x;	2
cout << " x = " << x;	x = 2

California State University Northridge 11

Review Input using cin

- Input prompt tells user what to input
- Enter several variables with a single cin command
  - Separate entries by a space and press <enter> (the Enter key) after last entry

Code	*Actions*
cout << "Enter x: "; cin >> x;	Type 2.3 <enter> for x = 2.3
cout << "Enter x, y, and z: "; cin >> x >> y >>z;	Type 1.4 -3.2 12.7 <enter> to set x = 1.4, y = -3.2 and z = 12.7

California State University Northridge 12

Review Output spacing

- cout does not provide any spacing between output or new lines
- E.g., x = 13.2; y = 12.6; cout << x << y; (or cout << x; cout << y;) would print 13.212.6
- You can put a string of blanks in your cout commands: cout << x << " " << y; would print 13.2 12.6

Review Escape Sequences

- Special characters to control printing entered in strings
  - \n for new line
  - \t for tab
  - \" for quotation mark
  - \\ for backslash
- E.g cout << x << "\t" << y << "\n";
- Or cout << x << "\t" << y << endl;

Review Output Examples

```
cout << "\n  radius = " << r <<
  "\ndiameter = " << d <<
  "\n    area = " << a;
```

```
cout << "\n  radius = " << r
     << "\ndiameter = " << d
     << "\n    area = " << a;
```

```
cout << endl;
cout << "  radius = " << r << endl;
cout << "diameter = " << d << endl;
cout << "    area = " << a;
```


Submitting Assignments

- Use a separate file, called submission file, that has all code and results
- Create project and source code file as usual then add this file to project
  - Source file: pex2.cpp
  - Submission file: pex2.txt
- Copy code from source file
- Copy output from screen

Submission File

- See detailed instructions in exercise two notes
- Select **Add New Item** from **Project** menu
- Choose a text file (.txt)
- Select file name like pex2.txt
- Type your name at top of blank file
- Copy code and screen output, as instructed, to this file

Copying the Screen

- See details in exercise two notes
  - Point mouse to top of output window
  - Right click mouse to get menu
  - Select **Edit** and **Mark** to get flashing block cursor in the upper-left-hand of window
  - Select the text that you want to copy
  - Hit the enter key
  - Move to window where you want the copied text and use paste command

Tasks for this Exercise

- Copy and paste task one code from assignment to IDE and run
- Copy code and screen input and output (for six cases) to submission file
- Task two is repeat of task one with variables changed from double to int
- Task three provides a single input command; run for one case only

Task One Code

```
#include <iostream>
using namespace std;
int main()
{
  double x, y, z;
  cout << "Enter a value for x: ";
  cin >> x;
  cout << "Enter a value for y: ";
  cin >> y;
  z = x / y;
  cout << "For x = " << x << " and y =
  " << y << " x / y = " << z << "\n\n";
  return EXIT_SUCCESS;
}
```


Entering Powers of Ten

- C++, like most languages, uses an "E" or "e" to enter powers of ten
  - 3.45×10^{12} is entered as 3.45e12
  - -6.19×10^{-32} is entered as –6.19e-32
- We write 10^n as shorthand for 1×10^n
- How do you enter 10^6 and 10^{-6}?
  - Enter $10^6 = 1 \times 10^6$ as 1e6, 1E6 or 1000000
  - Enter $10^{-6} = 1 \times 10^{-6}$ as 1e-6, 1E-6 or 0.000001; what are 10e6 or 10E-6?
  - $10e6 = 10 \times 10^6 = 10^7$; $10e\text{-}6 = 10 \times 10^{-6} = 10^{-5}$

Remaining Tasks

- Task four looks at simple output spacing in `cout` commands; one case only
- Task five: more complicated spacing assignment – write code giving output exactly as shown in exercise
- Tasks one and two show differences between `double` and `int` as well as effects of incorrect data
- Due this Thursday, February 9

Task Five Output

```
Results for programming exercise 2:

Larry Caretto        February 9, 2006

        Input value of x = 10
        Input value of y = 3
Output value of z = x / y = 3.33333

Press any key to continue
```

blank line
blank line
decimal result
blank line
Two Spaces
Note alignments