Chapter 3

The New Math

C++ Data Types

- integral
- floating
  - float
  - double
  - long double
- simple
- address
- structured
- pointer
  - array
  - struct
  - union
- reference
  - enum
Numeric Data Types

- char
  - 1 byte
- short
  - 2 bytes
- int
  - 4 bytes
- long
  - 4 bytes

More Numeric Data Types

- float
  - 4 bytes
- double
  - 8 bytes
- long double
  - 8 bytes
Declarations

- Constant Examples
  - `const float PI = 3.14159;`
  - `const float E = 2.71828;`
  - `const int MAX_SCORE = 100;`

- Variable Examples
  - `int studentCount;`
  - `char grade;`
  - `float averageGrade;`

---

Simple Arithmetic Expressions

- Expressions are made up of constants, variables and operators
- Examples
  - `alpha + 2`
  - `rate - 6.0`
  - `4 - alpha`
  - `rate`
  - `alpha * num`
Arithmetic Operators

- Unary Minus
- Unary Plus
- Subtraction
- Addition
- Multiplication
- Division
- Modulus

<table>
<thead>
<tr>
<th>Expression</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>3 + 6</td>
<td>9</td>
</tr>
<tr>
<td>3.4 - 6.1</td>
<td>-2.7</td>
</tr>
<tr>
<td>2 * 3</td>
<td>6</td>
</tr>
<tr>
<td>8 / 2</td>
<td>4</td>
</tr>
<tr>
<td>8.0 / 2.0</td>
<td>4.0</td>
</tr>
<tr>
<td>8 / 8</td>
<td>1</td>
</tr>
<tr>
<td>8 / 9</td>
<td>0</td>
</tr>
<tr>
<td>8 / 7</td>
<td>1</td>
</tr>
<tr>
<td>8 % 8</td>
<td>0</td>
</tr>
<tr>
<td>8 % 9</td>
<td>8</td>
</tr>
<tr>
<td>8 % 7</td>
<td>1</td>
</tr>
<tr>
<td>0 % 7</td>
<td>0</td>
</tr>
<tr>
<td>5 % 2.3</td>
<td>error</td>
</tr>
</tbody>
</table>
Oh yeah, don’t forget

- Increment
  - ++
- Decrement
  - --

Let’s Get Tricky

- Precedence Rules
  - Highest: ++ -- Unary + Unary –
  - Middle: * / %
  - Lowest: + -
  - See page 1056 Appendix B for complete list
Type Coercion and Type Casting

- **Type Coercion**
  - The implicit (automatic) conversion of a value from one data type to another

- **Type Casting**
  - The explicit conversion of a value from one data type to another

---

Examples

- float someFloat = 3.4 / 2;
- int someInt = 3.4 / 2;
- someFloat = 3 / 4;
- someInt = 3.4 / 2.2;
Casting

- `someFloat = (double) 3 / (double) 4;`
- `someFloat = <static_cast>(double) 3 / <static_cast> (double) 4;`

Function Calls

- Example
  - `int someInt = Cube(27);`
- Function Call (Function Invocation)
  - The mechanism that transfer control to the function
- Argument List (Parameter List)
  - A mechanism by which functions communicate with each other
Formatting Output

- `setw()`
  - Tells how many characters the next item outputted should have
- `fixed`
  - Force all subsequent floating-point output to appear in decimal notation
- `showpoint`
  - Force decimal points in all subsequent floating-point output; even whole numbers
- `setprecision()`
  - Tells how many decimal places all subsequent floating-point output should have

String Functions

- `length()` or `size()`
- `find()`
- `substr()`