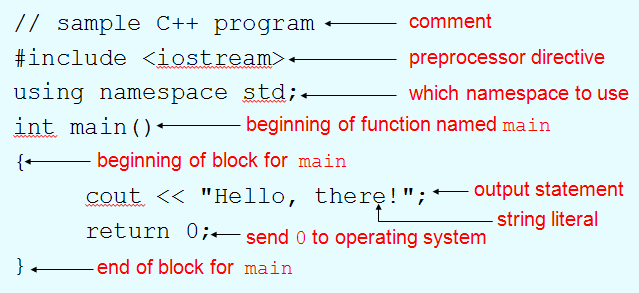
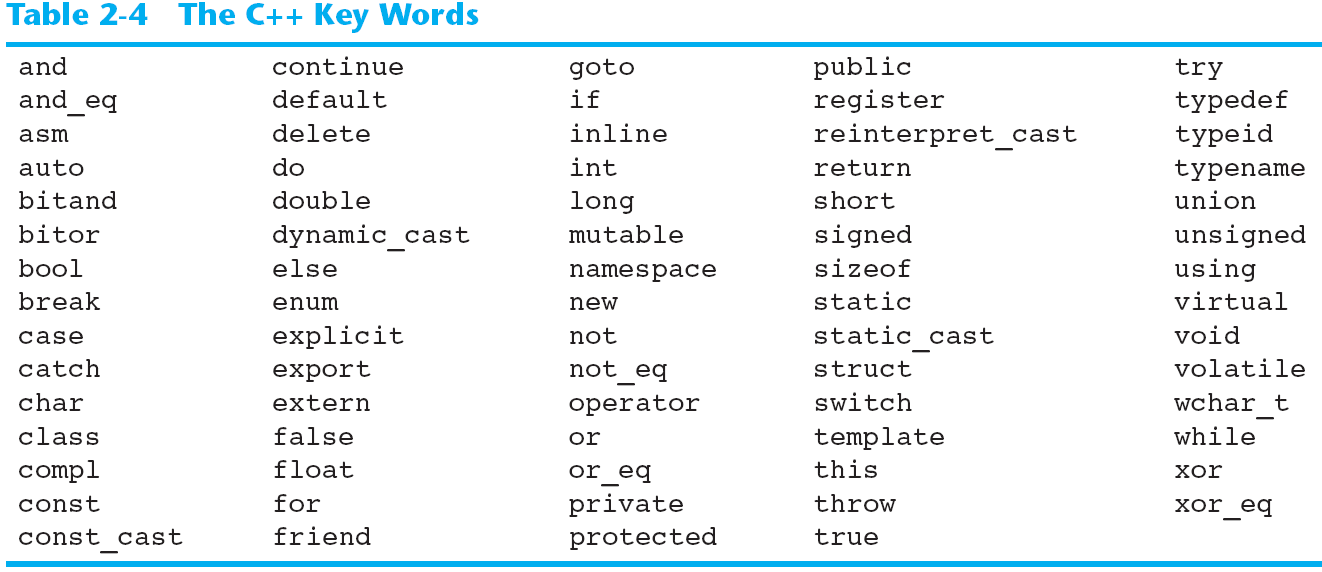
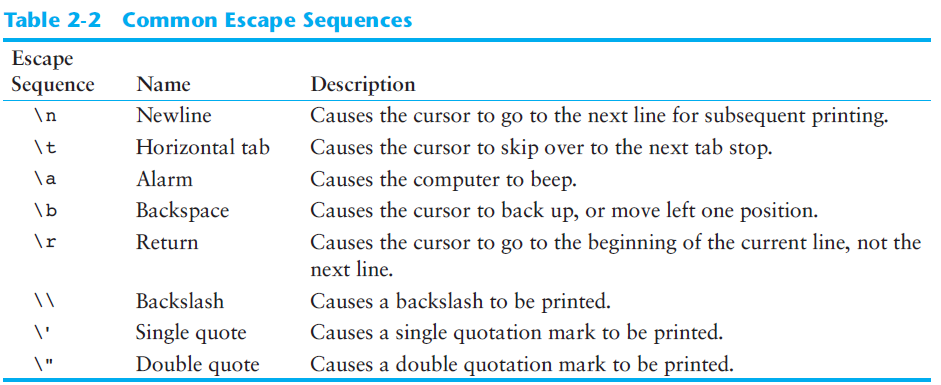
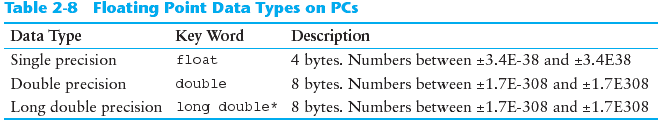
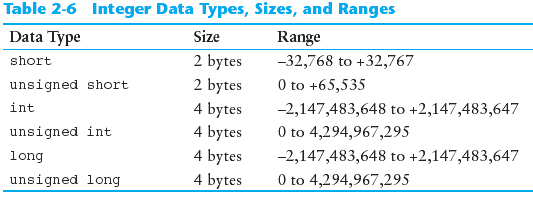
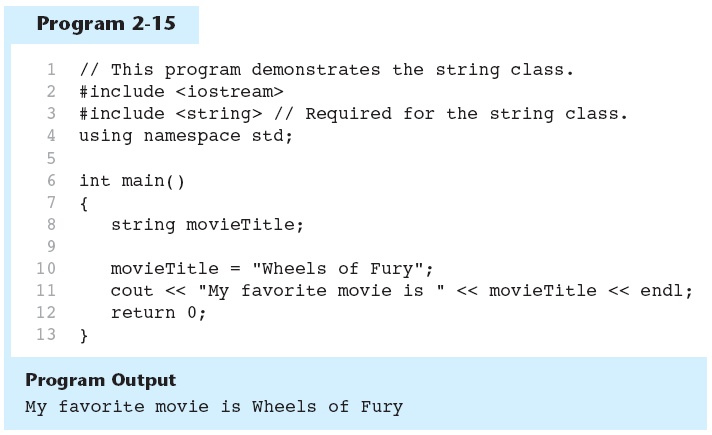
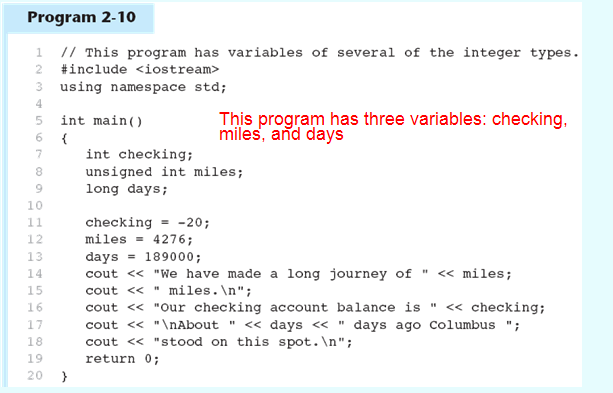
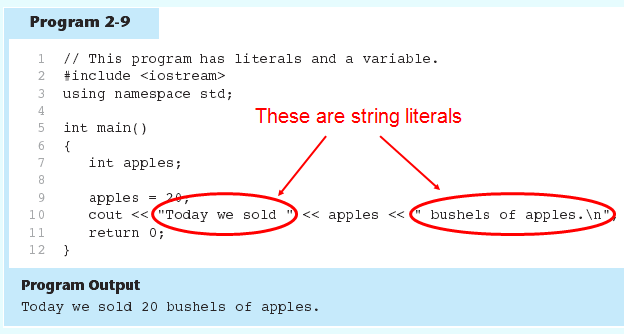
**Chapter 2: Introduction to C++**



|  |  |  |
| --- | --- | --- |
| Character | Name | Meaning |
| // | Double slash | Beginning of a comment |
| # | Pound sign | Beginning of preprocessor directive |
| < > | Open/close brackets | Enclose filename in #include |
| ( ) | Open/close parentheses | Used when naming a function |
| { } | Open/close brace | Encloses a group of statements |
| " " | Open/close quotation marks | Encloses string of characters |
| ; | Semicolon | End of a programming statement |



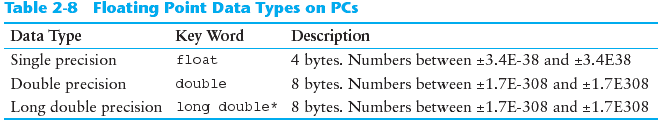


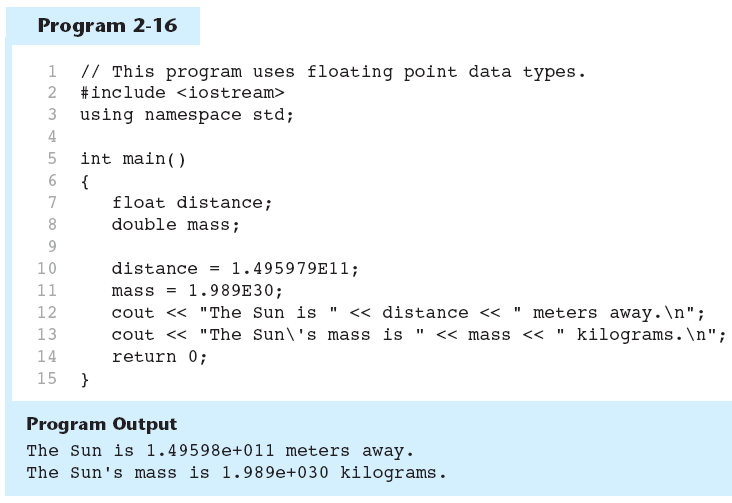


**Floating-point data types are used to de ne variables that can hold real numbers.**

Computers typically use *E notation* to represent floating-point values. In E notation, the number 47,281.97 would be 4.728197E4. The part of the number before the E is the mantissa, and the part after the E is the power of 10. When a floating point number is stored in memory, it is stored as the mantissa and the power of 10.

The floating-point data types are:  
**float  
double  
long double**





**Determining the Size of a Data Type**

The sizeof operator gives the size of any data type or variable:

double amount;

cout << "A double is stored in "

<< sizeof(double) << "bytes\n";

cout << "Variable amount is stored in "

<< sizeof(amount)

<< "bytes\n";

C++ has unary, binary, and ternary operators:

* unary (1 operand) -5
* binary (2 operands) 13 - 7
* ternary (3 operands) exp1 ? exp2 : exp3

