C++ Programming Basics

Ho Dac Hung

Basic Program Construction

```
#include <iostream>
using namespace std;
int main()
{
    cout << “Hello World\n”;
    return 0;
}
```

Comments

- Comment syntax: “//…” or “/*…*/”.
- When to use comments?

Variables

- Variables are the most fundamental part of any language. A variable has a symbolic name and can be given a variety of values.
- Variables are located in particular places in the computer's memory. When a variable is given a value, that value is actually placed in the memory space assigned to the variable.

Variable Declarations
Variable Definitions

- A declaration introduces a variable's name into a program and specifies its type.
- However, if a declaration also sets aside memory for the variable, it is also called a definition.

Variable Names

- The names given to variables (and other program features) are called identifiers.
- You can use upper and lowercase letters, and the digits from 1 to 9. You can also use the underscore (_).
- The first character must be a letter or underscore.
- Identifiers can be as long as you like, but most compilers will only recognize the first few hundred characters.
- You can’t use a C++ keyword as a variable name.

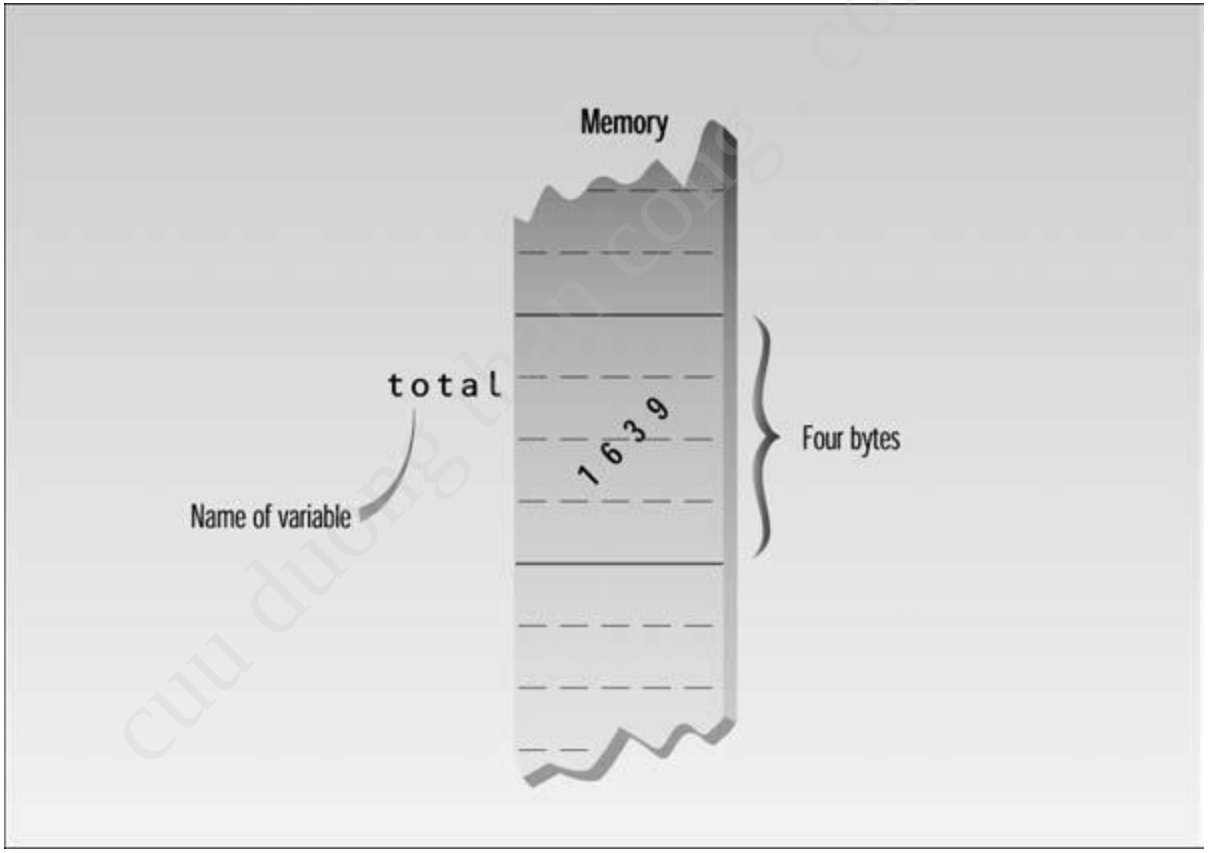
Assignment Statements

- The equal sign (=), as you might guess, causes the value on the right to be assigned to the variable on the left.

Integer Variables

- Integer variables exist in several sizes, but the most commonly used is type int. The amount of memory occupied by the integer types is system dependent. On a 32-bit system such as Windows, an int occupies 4 bytes (which is 32 bits) of memory.

Integer Variables



Integer Constants

- The number 20 is an integer constant. Constants don’t change during the course of the program. An integer constant consists of numerical digits. There must be no decimal point in an integer constant, and it must lie within the range of integers.

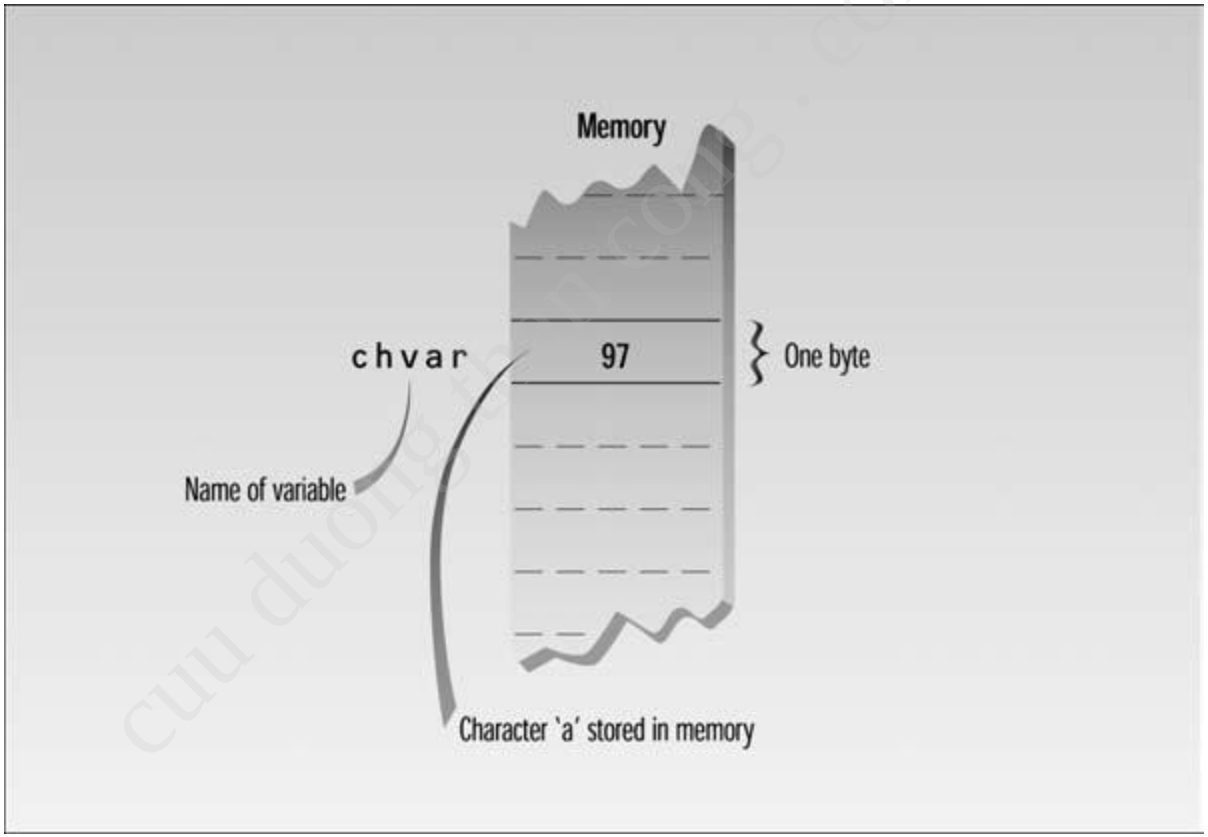
Other Integer Types

- The two most common types are long and types long and short have fixed sizes no matter what system is used.
- Type long always occupies four bytes. Type short occupies two bytes.

Character Variables

- Type char stores integers that range in value from –128 to 127. Variables of this type occupy only 1 byte (eight bits) of memory. Character variables are sometimes used to store numbers that confine themselves to this limited range, but they are much more commonly used to store ASCII characters.
- Standard C++ provides a larger character type called wchar_t (wchar.h) to handle foreign languages.

Character Variables



Character Constants

- Character constants use single quotation marks around a character, like ‘a’ and ‘b. When the C++ compiler encounters such a character constant, it translates it into the corresponding ASCII code. The constant ‘a’ appearing in a program, for example, will be translated into 97.

Escape Sequences

Escape Sequence	Character
\a	Bell
\b	Backspace
\n	Newline
\t	Tab
\\	Backslash
\’	Single quotation mark
\”	Double quotation marks
\xdd	Hexadecimal notation

Floating Point Types

- Floating-point variables represent numbers with a decimal place. Floating-point variables represent what mathematicians call real numbers.

Floating Point Types

- Type float stores numbers in the range of about 3.4×10^{-38} to 3.4×10^{38}, with a precision of 7 digits. It occupies 4 bytes (32 bits) in memory.

Floating Point Types

- Type double requires 8 bytes of storage and handles numbers in the range from 1.7×10^{-308} to 1.7×10^{308} with a precision of 15 digits. Type long double is compiler-dependent but is often the same as double.

Floating-Point Constants

- The number 3.14159F is an example of a floating-point constant. The decimal point signals that it is a floating-point constant, and not an integer, and the F specifies that it's type float, rather than double or long double. The number is written in normal decimal notation. You don't need a suffix letter with constants of type double; it's the default. With type long double, use the letter L.

Type bool

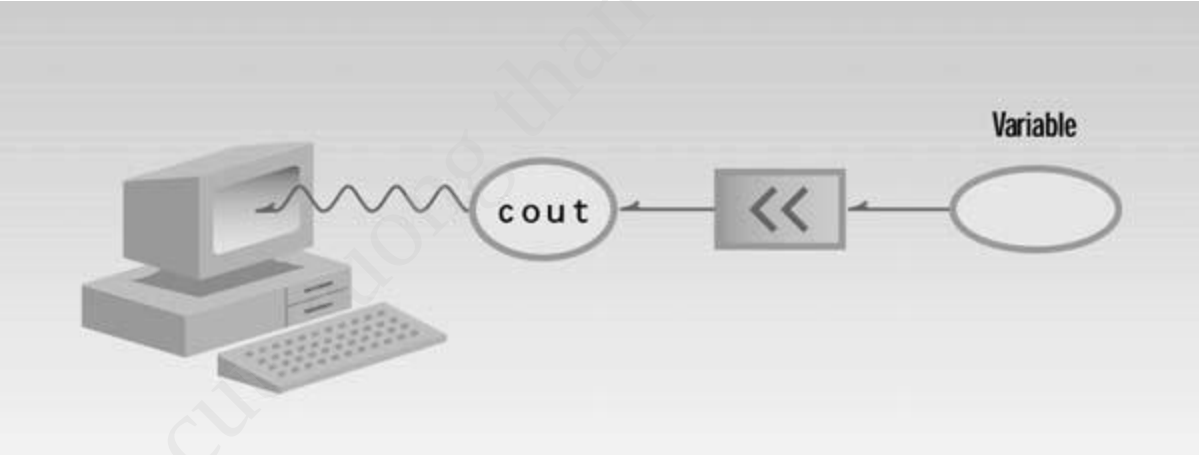
- Variables of type bool can have only two possible values: true and false. In theory a bool type requires only one bit (not byte) of storage, but in practice compilers often store them as bytes because a byte can be quickly accessed, while an individual bit must be extracted from a byte, which requires additional time.

Output Using cout

- The identifier cout is actually an object. It is predefined in C++ to correspond to the standard output stream. A stream is an abstraction that refers to a flow of data. The standard output stream normally flows to the screen display.
- The operator << is called the insertion or put to operator. It directs the contents of the variable on its right to the object on its left.

Output Using cout

```
cout << “var1+10 is “;
cout << var2 << endl;
```



Output Using cout

- Manipulators are instructions to the output stream that modify the output in various ways.
- The endl manipulators causes a linefeed to be inserted into the stream, so that subsequent text is displayed on the next line. It has the same effect as sending the '\n' character, but is somewhat clearer.
- The setw manipulator changes the field width of output.

Output Using cout

- The declarations for the manipulators (except endl) are not in the usual iostream header file, but in a separate header file called iomanip.

Input with cin

- The keyword cin (pronounced “C in”) is an object, predefined in C++ to correspond to the standard input stream. This stream represents data coming from the keyboard (unless it has been redirected). The >> is the extraction or get from operator. It takes the value from the stream object on its left and places it in the variable on its right.

The const Qualifier

- The keyword const (for constant) precedes the data type of a variable. It specifies that the value of a variable will not change throughout the program. Any attempt to alter the value of a variable defined with this qualifier will elicit an error message from the compiler.

The #define Directive

- constants can also be specified using the preprocessor directive #define. This construction has long been popular in C. However, you can't specify the data type of the constant using #define, which can lead to program bugs.

Variable Type Summary

- bool
- char
- short
- int
- long
- float
- double
- long double

unsigned Data Types

- unsigned char
- unsigned short
- unsigned int
- unsigned long

Type Conversion

- C++, like C, is more forgiving than some languages in the way it treats expressions involving several different data types.
- When two operands of different types are encountered in the same expression, the lower-type variable is converted to the type of the higher-type variable.

Type Conversion

- long double
- double
- float
- long
- int
- short
- char

Type Conversion

- Sometimes a programmer needs to convert a value from one type to another in a situation where the compiler will not do it automatically or without complaining.

```
aCharVar = static_cast<char>(anIntVar);
```

Arithmetic Operators

- +
- -
- *
- /
- %
- +=
- -=
- *=
- /=
- %=
- ++
- --

Arithmetic Functions

- cos
- sin
- tan
- acos
- asin
- atan
- atan2
- exp
- log
- log2
- log10
- pow
- sqrt
- ceil
- floor
- round
- trunc

Library Functions

- Many activities in C++ are carried out by library functions. These functions perform file access, mathematical computations, and data conversion, among other things.

Library Functions

- To use a library function, you must link the library file that contains it to your program. The appropriate functions from the library file are then connected to your program by the linker.
- However, that's not the end of the story. The functions in your source file need to know the names and types of the functions and other elements in the library file. They are given this information in a header file. Each header file contains information for a particular group of functions.

Library Functions