

Chapter 1 - Introduction to Computer Programming

(计算机导论)

Quick Quiz 1

1. The smallest and most basic data item in a computer is a **bit**; it is really a switch that can be either open (0) or closed (1).
2. What is the ALU of a computer?
The Arithmetic and Logic Unit (ALU) of a computer performs all of the computations, such as addition, subtraction, comparisons, and so on, that a computer provides.
3. What is the control unit of a computer?
The control unit of a computer directs and monitors the overall operation of the computer.
4. A(n) **direct access storage device (DASD)** allows a computer to read or write any one file or program independent of its position on the storage medium.

Quick Quiz 2

1. What is an assembly language?
Programming languages that use the substitution of word-like symbols, such as ADD, SUB, MUL, for the binary opcodes, and both decimal numbers and labels for memory addresses are referred to as assembly languages.
2. The program that translates a high-level source program as a complete unit before any individual statement is executed is called a(n) **compiler**.
3. What is a linker?
A linker combines additional machine language code with the object program to create a final executable program.
4. When English phrases are used to describe an algorithm (the processing steps), the description is called **pseudocode**.

Quick Quiz 3

1. What is the software development process?
The technique used by professional software developers for understanding the problem that is being solved and for creating an effective and appropriate software solution is called the software development process.
2. When writing a program, a(n) **selection** structure provides the capability to make a choice between different instructions, depending on the result of some condition.
3. When writing a program, a(n) **invocation** structure involves summoning into action specific sections of code as they are needed.
4. What is a repetition structure?
When writing a program, a repetition structure, which is also referred to as looping and iteration, provides the ability for the same operation to be repeated based on the value of a condition.

Key Terms

- **Application software** consists of programs written to perform particular tasks required by users.
- The **Arithmetic and Logic Unit (ALU)** of a computer performs all of the computations, such as addition, subtraction, comparisons, and so on, that a computer provides.
- Translator programs that translate assembly language programs into machine language programs are known as **assemblers**.
- Programming languages that use the substitution of word-like symbols, such as ADD, SUB, MUL, for the binary opcodes, and both decimal numbers and labels for memory addresses are referred to as **assembly languages**.
- The smallest and most basic data item in a computer is a **bit**; it is really a switch that can be either open (0) or closed (1).
- The **bootstrap loader** is internally contained in ROM and is a permanent, automatically executed component of the computer's system software.
- The grouping of 8 bits to form a larger unit is an almost universal computer standard and is referred to as a **byte**.
- The collections of patterns consisting of 0s and 1s used to represent letters, single digits, and other single characters are called **character codes**.
- Converting an algorithm into a computer program, using a language such as C, is called **coding the algorithm**.
- When all of the statements in a high-level source program are translated as a complete unit before any individual statement is executed, the programming language is called a **compiled language**.
- The program that translates a high-level source program as a complete unit before any individual statement is executed is called a **compiler**.
- A **computer program** is a structured combination of data and instructions that is used to operate a computer and produce a specific result.
- The **control unit** of a computer directs and monitors the overall operation of the computer.
- A **direct access storage device (DASD)** allows a computer to read or write any one file or program independent of its position on the storage medium.
- An **executable program** is a program that can operate a computer.
- A **first-level structure diagram** for an algorithm represents the first attempt at an initial, but not yet sufficiently detailed, structure for a solution algorithm.
- Initially, the most common magnetic disk storage device was the removable **floppy disk**(软盘).
- A **flowchart** provides a pictorial representation of an algorithm using specifically defined shapes.
- When mathematical equations are used to describe an algorithm, the description is called a **formula**.
- In C, a procedure is referred to as a **function**.

- Collectively, the components used to make a computer are referred to as **hardware**.
- When each statement in a high-level source program is translated individually and executed immediately upon translation, the programming language is called an **interpreted language**.
- The program that translates each statement in a high-level source program and executes it immediately upon translation is called an **interpreter**.
- When writing a program, an **invocation** structure involves invoking, or summoning into action, specific sections of code as they are needed.
- A **linker** combines additional machine language code with the object program to create a final executable program.
- Both machine and assembly languages are classified as **low-level languages**; this is because both of these language types use instructions that are directly tied to one type of computer.
- Executable programs are always written as a sequence of binary numbers, which is a computer's internal language, and are also referred to as **machine language programs**.
- A **magnetic hard disk** consists of either a single rigid platter or several platters that spin together on a common spindle.
- In Java, a procedure is referred to as a **method**.
- CPUs are constructed as a single microchip, which is referred to as a **microprocessor**.
- Operating systems that permit each user to run multiple programs are referred to as both **multiprogrammed** and **multitasking** systems.
- **Multiuser systems** are able to handle multiple users concurrently.
- The output produced by the compiler is called an **object program**, which is a machine language version of the source code.
- Languages with object orientation like C++, Java, Visual Basic, and C#, are known as **object-oriented languages**.
- **Opcodes** is short for operation code.
- Collectively, the set of system programs used to operate and control a computer is called the **operating system**.
- The three tasks of an overall solution algorithm are the primary responsibility of almost every problem, and we refer to this algorithm as the **Problem-Solver Algorithm**.
- In a **procedural language**, the available instructions are used to create self-contained units, referred to as procedures.
- The purpose of a **procedure** is to accept data as input and transform the data in some manner to produce a specific result as an output.
- The program instructions resulting from coding an algorithm are referred to as **program code**, or simply **code**, for short.
- In an automobile, control is provided by the driver, who sits inside of and directs the car; in a computer, the driver is called a **program**.
- A statement of a problem, or a specific request for a program, is referred to as a

program requirement.

- The set of instructions that can be used to construct a program is called a **programming language**.
- **Programming** is the process of writing instructions in a language that the computer can respond to and that other programmers can understand.
- When English phrases are used to describe an algorithm (the processing steps), the description is called **pseudocode**.
- When writing a program, a **repetition** structure, which is also referred to as **looping** and **iteration**, provides the ability for the same operation to be repeated based on the value of a condition.
- Each field of study has its own name for the systematic method used to design solutions to problems. In science this method is referred to as the **scientific method**, while in engineering disciplines the method is referred to as the **systems approach**.
- When writing a program, a **selection** structure provides the capability to make a choice between different instructions, depending on the result of some condition.
- When writing a program, a **sequence** structure defines the order in which instructions are executed by the program.
- Another term for a program or set of programs is **software**.
- The technique used by professional software developers for understanding the problem that is being solved and for creating an effective and appropriate software solution is called the **software development process**.
- Programs written in a computer language (high or low level) are referred to interchangeably as both **source programs** and **source code**.
- A **structured language** is a high-level procedural language, such as C, that enforces structured procedures.
- Procedures conforming to structure guidelines are known as **structured procedures**.
- **System software** is the collection of programs that must be readily available to any computer system to enable the computer to operate.
- A **top-down algorithm development** starts at the topmost level and proceeds to develop more and more detailed algorithms as it proceeds to the final set of algorithms.
- Main memory is **volatile**; whatever is stored in it is lost when the computer's power is turned off.
- Main memories combine 1 or more bytes into a single unit, referred to as a **word**.

Chapter 2 - Getting Started in C Programming（标识符、头文件、基础数据类型、运算符、占位符）

Quick Quiz 1

1. What is an identifier?

The names of functions, as well as all of the words permitted in a program that have

special meaning to the compiler, are collectively referred to as identifiers.

2. What is a function header line?

A function header line, which is always the first line of a function, contains three pieces of information (1) what type of data, if any, is returned by the function, (2) the name of the function, and (3) what type of data, if any, is sent into the function.

3. The two characters \ and n, when used together, are called a(n) **newline escape sequence**.
4. A(n) **reserved word/keyword** is a word that is predefined by the programming language for a special purpose and can only be used in a specified manner for its intended purpose.

Quick Quiz 2

1. What is a data type?

A data type is defined as a set of values and a set of operations that can be applied to these values.

2. In numerical theory, the term **precision** typically refers to numerical accuracy.

3. What is an expression?

An expression is any combination of operators and operands that can be evaluated to yield a value.

4. **Associativity** is the order in which operators of the same precedence are evaluated.

Quick Quiz 3

1. **Variables** are simply names given by programmers to computer storage locations.

2. What is an assignment statement?

An assignment statement tells the computer to assign a value to (that is, store a value in) a variable.

3. What is a definition statement?

Definition statements define or tell the compiler how much memory is needed for data storage.

3. When a declaration statement provides an initial value, the variable is said to be **initialized**.

Key Terms

- Items passed to a function are always placed within the function name parentheses(括号) and are called **arguments**.
- Data transmitted into a function at run time are referred to as **arguments of the function**.
- The operators used for arithmetic operations are called **arithmetic operators**.
- An **assignment statement**(赋值语句) tells the computer to assign a value to (that is, store a value in) a variable.
- **Associativity**(运算符结合律) is the order in which operators of the same precedence are evaluated.

- **Binary operators** require two operands to produce a result.
- A **built-in data type** is one that is provided as an integral part of the language.
- A **comment**(注释) is a note about the code that the programmer includes so that he (or other programmers) can keep track of what the various parts of the program do.
- A control string is referred to as a **control specifier**(控制字符串).
- A string that also includes a **conversion control sequence**(转换控制序列), such as %d, is termed a **control string**.
- Conversion control sequences are also referred to as **conversion specifications** and **format specifiers**(格式化占位符).
- A **data type** is defined as a set of values *and* a set of operations that can be applied to these values.
- **Definition statements** define or tell the compiler how much memory is needed for data storage.
- A `double` value is sometimes referred to as a **double-precision** number.
- The `main()` function is sometimes referred to as a **driver function**(驱动函数), because it tells the other functions the sequence in which they are to operate.
- The backslash character, \, is also known as the **escape character**(转义字符).
- The combination of a backslash and one of several specific characters is called an **escape sequence**(转义序列).
- All statements that cause some specific action to be performed by the computer when the function is executed must end with a semicolon (;)(必须以分号隔开语句); such statements are known as **executable statements**(可执行语句).
- An **expression** is any combination of operators and operands that can be evaluated to yield a value.
- An expression containing only floating-point values (single and double precision) as operands is called a **floating-point expression** (the term **real expression** is also used), and the result of such an expression is a double-precision value.
- A **floating-point value**, which is also called a **real number**, can be the number zero or any positive or negative number that contains a decimal point.
- A **function header line**, which is always the first line of a function, contains three pieces of information:
 - (1) what type of data, if any, is returned by the function.
 - (2) the name of the function.
 - (3) what type of data, if any, is sent into the function.
- A **header file** is placed at the top, or head, of a C program using the `#include` command.
- The names of functions, as well as all of the words permitted in a program that have special meaning to the compiler, such as `radius` and `circumference`, are collectively referred to as **identifiers**(标识符).
- Declaration statements can also be used to store an initial value into declared variables; this value is referred to as an **initial value**(初始化值).
- When a declaration statement provides an initial value, the variable is said to be **initialized**.

- **Invoking** a function is more commonly referred to as **calling the function**.
- Reserved words are also referred to as **keywords** in C.
- A **literal**(字面值) is an acceptable value for a data type.
- Another name for a literal is a **literal value**(字面值), or **constant**.
- An expression containing both integer and floating-point values is called a **mixed-mode expression**(混合模式表达式——有整数和浮点数的表达式).
- The % operator, called both the **modulus** and **remainder operator**(余数运算符), captures the remainder when an integer number is divided by an integer.
- Under no circumstances may comments be **nested**(嵌套, 不能嵌套注释)—one comment containing another comment.
- The two characters \ and n, when used together, are called a **newline escape sequence**(换行转义序列).
- An **operand** can be either a literal value or an identifier that has a value associated with it. (操作数可以是文字字符也可以是其对应的值——ASCII)
- Inputting data or messages to a function is called **passing data to the function**.
- In numerical theory, the term **precision**(精度) typically refers to numerical accuracy.
- Built-in types are also known as **primitive types**(基本类型).
- A large number of the identifiers used in a C program are selected by the programmer, and are known as **programmer-created identifiers**(程序员创建的标识符) or **programmer-created names**.
- The keywords `short`, `long`, and `unsigned` are known as **qualifiers**(限定符), because they qualify the meaning of the keyword `int`.
- A **reserved word**(保留字) is a word that is predefined by the programming language for a special purpose and can only be used in a specified manner for its intended purpose.
- `short int`, `int`, and `long int` data types are formally referred to as **signed data types**(有符号数据类型——正与负).
- A **simple binary arithmetic expression** consists of a binary arithmetic operator connecting two literal values in the form: `literalValue operator literalValue`.
- A `float` value is sometimes referred to as a **single-precision** number.
- **Standard identifiers**(标准标识符) are words that are predefined(预定义) in C.
- Messages are known as **strings** in C, because they consist of a string of characters made up of letters, numbers, and special characters.
- A programming language's **syntax**(句法/语法) is the set of rules for formulating statements that are “grammatically correct” for the language.
- A **unary operator**(一元操作符) is one that operates on a single operand.
- An **unsigned data type** provides only for nonnegative (that is, zero and positive) values.
- The address of the first memory byte used for storing a variable is known as the **variable's address**.

- **Variables** are simply names given by programmers to computer storage locations.
- In C, **white space** refers to any combination of one or more blank spaces, tabs, or new lines.(空格指的是一个或多个空格、制表符或新行的任意组合)

Chapter 3 - Processing and Interactive Input(面向过程、交互语句)

Quick Quiz 1

1. In C, the **=** symbol is called the assignment operator.
2. The automatic conversion across an assignment operator is referred to as a(n) *implicit* type conversion.
3. What is a garbage value?
A previously stored number, if it has not been initialized to a specific and known value, is frequently referred to as a garbage value.(没有被初始化为指定值就是垃圾值)
4. What is the prefix increment operator?
Using the increment operator, ++, the expression variable = variable + 1 can be replaced by the either the expression variable++ or ++variable. When the ++ operator appears before a variable, it is called a prefix increment operator.

Quick Quiz 2

1. A(n) *prompt(提示符)* is a message that tells the person at the screen what should be typed.
2. On most computer systems, characters read by the keyboard are stored in a temporary holding area called a(n) *buffer* immediately after they are pressed.
3. What are robust programs?
Programs that detect and respond effectively to unexpected user input are formally referred to as robust(健全的) programs and informally as “bullet-proof(防弹的)” programs.
4. What is user-input validation?
The basic approach to handling invalid data input is referred to as user-input validation, which means validating the entered data either during or immediately after the data have been entered, and then providing the user with a way of reentering any invalid data.

Quick Quiz 3

1. The format of numbers displayed by `printf()` can be controlled by *field width specifiers* included as part of each conversion control sequence.
2. What are magic numbers?
Literal values that appear many times in the same program are referred to by programmers as magic numbers. (被引用多次的值被称为幻数?)
3. `#define` statements are also called *equivalence(等价)* statements.

4. What does the term “literal data” mean?

Literal data refers to any data within a program that explicitly identifies itself. (字面数据指的是程序中显式标识自身的任何数据。)

Key Terms

- User-defined data types are formally referred to as **abstract data types**(用户定义的数据类型被称为抽象数据类型).
- In C, the = symbol is called the **assignment operator**(赋值运算符).
- On most computer systems, characters read by the keyboard are stored in a temporary holding area called a **buffer**(缓冲区) immediately after they are pressed.
- The operator used to force the conversion of a value to another type is the **cast**(强制转换) operator.
- A special type of assignment statement that is very similar to the accumulating statement is the **counting statement**(计数语句?).
- #define statements are also called **equivalence** statements.
- The format of numbers displayed by printf() can be controlled by **field width specifiers**(字符宽度标识) included as part of each conversion control sequence.
- A previously stored number, if it has not been initialized to a specific and known value, is frequently referred to as a **garbage value**.
- The automatic conversion across an assignment operator is referred to as an **implicit type conversion**(隐式类型转换).
- Using the **increment operator**(增量运算符), ++, the expression `variable = variable + 1` can be replaced by either the expression `variable++` or `++variable`.
- **Literal data** refers to any data within a program that explicitly identifies itself.
- The term **lvalue**(左值) refers to any quantity that is valid on the left side of an assignment operator.
- Literal values that appear many times in the same program are referred to by programmers as **magic numbers**.
- When the -- operator appears after a variable, it is called a **postfix decrement operator**(后自减运算符).
- When the ++ operator appears after a variable, it is called a **postfix increment operator**.
- When the -- operator appears before a variable, it is called a **prefix decrement operator**.
- When the ++ operator appears before a variable, it is called a **prefix increment operator**.
- The assigning of a name to a function or procedure in such a way that the function is invoked by simply using a name with appropriate arguments is formally referred to as **procedural abstraction**(过程抽象).
- A **prompt** is a message that tells the person at the screen what should be typed.

- Programs that detect and respond effectively to unexpected user input are formally referred to as **robust** programs and informally as “bullet-proof” programs.
- An **rvalue**(右值) refers to any quantity that is valid on the right side of an assignment operator.
- Other terms for symbolic names are **symbolic constants**(符号常量) and **named constants**(命名常量).
- C provides the programmer with the capability to define a value (that will be used throughout a program) once by equating the number to a **symbolic name**(就是 **#define** 预定义的常量值).
- The basic approach to handling invalid data input is referred to as **user-input validation**, which means validating the entered data either during or immediately after the data have been entered, and then providing the user with a way of reentering any invalid data.
- The term **validate**(验证) means checking that the entered value matches the data type of the variable that the value is assigned to within a `scanf()` function call, and that the value is within an acceptable range of values appropriate to the application.

Chapter 4 - Selection(条件判断语句)

Quick Quiz 1

1. What is “flow of control”?
The term flow of control refers to the order in which a program’s statements are executed.(程序语句执行的顺序)
2. A(n) **relational** expression consists of a relational operator that compares two operands. (条件判断表达式)
3. Relational expressions are also known as **conditions**.
4. How does the NOT (!) operator work?
If an expression has any non-0 value (true), !expression produces a 0 value (false). If an expression is false to begin with (has a 0 value), !expression is true and evaluates to 1.

Quick Quiz 2

1. The simplest C selection statement(选择/判断语句) is the **one-way**(单向的) `if` statement.
2. A(n) **compound**(复合) statement is one or more statements contained between braces.
3. What is a nested(嵌套的) `if` statement?
Including one or more `if-else` statements within an `if` or `if-else` statement is referred to as a nested `if` statement.
4. Is indentation(缩进) important in the evaluation of `if-else` statements?

*Indentation used within an **if-else** is always irrelevant as far as the compiler is concerned. Whether the indentation exists or not, the compiler will, by default, associate an **else** with the closest previous unpaired **if**, unless braces are used to alter this default pairing. (无关紧要就是了)*

Quick Quiz 3

1. What is a `switch` statement?

*A **switch** statement is a specialized selection statement that can be used in place of an **if-else** chain where exact equality to one or more integer constants is required.*

2. Internal to the `switch` statement, the keyword **case** identifies the values that will be compared to the value of the `switch` expression.

3. What is the role of the `default` statement in a `switch`?

*Any number of **case** labels may be contained within a **switch** statement, in any order. However, if the value of the expression does not match any of the **case** values, no statement is executed unless the keyword **default** is encountered. The word **default** is optional and operates the same as the last **else** in an **if-else** chain. If the value of the expression does not match any of the **case** values, program execution begins with the statement following the word **default***

4. Once an entry point has been located by the `switch` statement, no further `case` evaluations are done; this means that unless a(n) **break** statement is encountered, all statements that follow, until the closing brace of the `switch` statement, will be executed.

Key Terms

- A **compound statement** is one or more statements contained between braces.
- Relational expressions are also known as **conditions**.
- A **debugger** program controls the execution of a C program, can interrupt the C program at any point in its execution, and can display the values of all variables at the point of interruption.
- In computer jargon, a program error is referred to as a **bug**, and the process of isolating, correcting, and verifying the correction is called **debugging**.
- **Defensive programming**(防御性编程) is a technique where the program includes code to check for improper data before an attempt is made to process it further. (检查非法数据)
- **Diagnostic `printf()` statements**(诊断性 `printf` 语句) can be a considerable help in debugging.
- **Echo printing**(回显打印) is the technique to add temporary code that displays the values of all input data.
- The term **flow of control**(控制流) refers to the order in which a program's statements are executed.

- A nested(嵌套) `if` construction, in which each nested `if` is written in the same line as the previous `else`, is called an **if-else chain**, and is used extensively in many programming problems.
- The defensive programming technique of checking user input data for erroneous or unreasonable data is referred to as **input data validation**.
- Including one or more `if-else` statements within an `if` or `if-else` statement is referred to as a **nested if statement**.
- The simplest C selection statement is the **one-way if statement**.
- **Program tracing(程序跟踪)** is the technique to imitate the computer and execute each statement by hand, as the computer would. (其实就是单步调试)
- A **relational expression(条件判断表达式)** consists of a relational operator that compares two operands.
- **Short-circuit evaluation(短路求值)** is the feature for the `&&` and `||` operators that makes the evaluation of an expression stop as soon as it is determined that an expression is false.

Chapter 5 - Repetition(循环语句)

Quick Quiz 1

1. What is a loop?
A section of code that is repeated is referred to as a loop, because after the last statement in the code is executed, the program branches, or loops, back to the first statement and starts another repetition through the code.
2. In a(n) **counter-controlled(计数控制循环)** loop, which is also known as a fixed-count loop, the condition is used to keep track of the number of repetitions that have occurred.
3. What is a program loop?
*The transfer of control back to the start of a **while** statement to reevaluate the expression is known as a program loop.*
4. A(n) **sentinel-controlled(标记控制型)** loop is a condition-controlled loop where one specific value is required to terminate the loop. (其实指的就是 `while` 循环)

Quick Quiz 2

1. In computer programming, data values used to signal either the start or end of a data series are called **sentinels(哨兵? 标记?)**.
2. On IBM-compatible computers, the EOF mark is generated whenever the **Ctrl+Z** keys are pressed simultaneously.
3. How does a `break` statement work?
*A **break** statement, as its name implies, forces an immediate break, or exit, from switch, while, for, and do-while statements only.*
4. What happens if you omit(省略) the tested expression in a `for` loop?

Although the initializing and altering lists can be omitted from a `for` statement, omitting the tested expression results in an infinite loop.

Quick Quiz 3

1. What is a nested loop?

There are many situations in which it is very convenient to have a loop contained within another loop. Such loops are called nested loops.

2. The second loop of a nested loop is called the **inner** loop.

3. A(n) **do-while** statement always creates a posttest(后测试/后判断) loop.

4. What type of application is ideally suited for a posttest loop?

There is one type of application that is ideally suited for a posttest loop, which is the input data validation application. (输入数据后再进行验证的程序)

Key Terms

- Lists in C, where commas are required to separate individual expressions in the list, are referred to as **comma-separated**(逗号分隔) lists.
- In a **condition-controlled loop**(条件控制型循环), the tested condition does not depend on a count being achieved, but rather on a specific value being encountered. (不取决于计数器, 而取决于达到其他的特定条件)
- In a **counter-controlled loop**(计数控制型循环), which is also known as a **fixed-count loop**, the condition is used to keep track of the number of repetitions that have occurred.
- Pretest(预测试) loops are also referred to as **entrance-controlled loops**(入口控制型循环).
- The second loop of a nested loop is called the **inner loop**.
- The **input data validation application** is ideally suited for a posttest loop.
- Each repetition in a loop is referred to as an **iteration**(迭代) or **pass through the loop**(通过了循环).
- A section of code that is repeated is referred to as a **loop**, because after the last statement in the code is executed, the program branches, or loops, back to the first statement and starts another repetition through the code.
- **Nested loops** have a loop contained within another loop.
- A **null statement**(空语句) is a do-nothing statement that is used where a statement is syntactically required, but no action is called for. (没用但有必要写的语句)
- The first loop of a nested loop is called the **outer loop**(外层循环).
- A loop that evaluates a condition at the end of the repeating section of code is referred to as a **posttest loop** or **exit-controlled loop**(出口控制循环).
- This type of loop is referred to as a **pretest loop** because the condition is tested before any statements within the loop are executed.
- In computer programming, data values used to signal either the start or end of a data series are called **sentinels**(哨兵/标记?).

Chapter 6 - Modularity Using Functions: Part I(函数的简介、函数头函数定义函数声明、参数值传递)

Quick Quiz 1

1. What is the difference between a called function and a calling function?
A function that is called or summoned into action by its reference in another function is a called function. A function that calls another function is referred to as the calling function.
2. The items enclosed within the parentheses in a function call statement are called *arguments/actual arguments/actual parameters* of the function.
3. What is a pass by value?
When a function simply receives copies of the values of each of the arguments and must determine where to store these values before it does anything else, this is known as a pass by value (or a call by value).
4. The argument names in the header line of a function are known as *parameters/formal parameters/formal arguments*.

Quick Quiz 2

1. A(n) *stub(存根)* is the beginning of a final function that is used as a placeholder for the final function until the function is completed. (如果一个函数体内没有任何语句，那么这个函数就可以被称为存根，能够为未实现的代码占一个位置)
2. Pass by value is also referred to as *call by value*.
3. How can you return a value from a function?
To return a value you use a return statement. For example, return expression; or return (expression);
4. How do you write the prototype of a function with an empty parameter list?
The prototype for a function with an empty parameter list requires either writing the keyword void or nothing at all between the parentheses(括号) following the function's name.

Quick Quiz 3

1. What are random numbers?
Random numbers are a series of numbers whose order cannot be predicted.
2. What are pseudorandom numbers?
Pseudorandom numbers are numbers which are not really random, but are sufficiently random for the task at hand.
3. The method for adjusting the random numbers produced by a random-number generator to reside within a specified range is called *scaling(缩放)*.
4. The standard library consists of **15** header files.

Key Terms

- Other terms used as synonyms for arguments are **actual arguments** and **actual parameters**.
- A fairly common procedure in child development is to establish normal ranges for height and weight as they relate to a child's age; these normal ranges are frequently referred to as **age norms**(年龄标准???你跟 C 有什么关系???).
- The items enclosed within the parentheses in a function call statement are called **arguments** of the function.
- A function that is called or summoned into action by its reference in another function is a **called function**.
- A function that calls another function is referred to as the **calling function**.
- The purpose of a **function body**(函数体) is to operate on the passed data and return, at most, one value directly back to the calling function.
- The portion of the function header that contains the function name and parameters is known as a **function declarator**(函数声明), which should not be confused with a function declaration (prototype 是函数原型).
- The purpose of a **function header**(函数头) is to identify the data type of the value returned by the function, if any, provide the function with a name, and specify the number, order, and type of values expected by the function. (比函数声明多了一个返回值类型的说明)
- A **function prototype** declares the function to the compiler—it tells the compiler the name of the function, the data type of the value that the function will return (the keyword `void` indicates that the function will not be returning any value), and the data types of each argument that the function expects to receive when it is called.
- The argument names in the header line of a function are known as **parameters** or **formal parameters** and **formal arguments**.
- When a function simply receives copies of the values of each of the arguments and must determine where to store these values before it does anything else, this is known as a **pass by value** (or a **call by value**).
- **Pseudorandom numbers** are numbers which are not really random, but are sufficiently random for the task at hand.
- **Random numbers** are a series of numbers whose order cannot be predicted.
- The method for adjusting the random numbers produced by a random-number generator to reside within a specified range is called **scaling**.
- A **stub** is the beginning of a final function that is used as a placeholder for the final function until the function is completed.

Chapter 7 - Modularity Using Functions: Part II(函数的作用域、局部变量、参数地址传递、递归函数)

Quick Quiz 1

1. What are local variables?

Variables that are created inside a function and available only to the function are said to be local to the function, or local variables.

2. What is scope?

Scope(作用域) is defined as the section of the program where the variable is valid or “known.”

3. A variable with a(n) **local** scope is simply one that has had storage locations set aside for it by a declaration statement made within a function body.

4. A variable with **global** scope is one whose storage has been created for it by a declaration statement located outside any function.

Quick Quiz 2

1. Where and how long a variable’s storage locations are kept before they are released can be determined by the **storage class** of the variable.

2. What are the registers in a computer?

Registers are high-speed storage areas physically located in the computer’s processing unit.

3. What is pass by reference(引用传递)?

Passing an address is referred to as a function pass by reference, because the called function can reference, or access, the variable using the passed address.(函数可以引用变量)

4. A variable that can store an address is known as a(n) **pointer variable/pointer**.

Quick Quiz 3

1. Functions that call themselves are referred to as self-referential(自引用) or **recursive(递归)** functions.

2. When a function invokes itself, the process is called **direct(直接)** recursion.

3. What is mutual recursion(互递归)?

A function can invoke a second function, which in turn invokes the first function; this type of recursion is referred to as indirect(间接) or mutual(相互) recursion.

4. With respect to(关于……) computer program execution, what is the stack?

C allocates new memory locations for all function arguments and local variables as each function is called. This allocation is made dynamically, as a program is executed, in a memory area referred to as the stack. (C 为运行中的每个函数内的局部变量动态分配内存空间，这部分空间就是栈区)

Key Terms

- With respect to storage classes, the term `auto` is short for **automatic**.
- When a function invokes itself, the process is called **direct recursion**.
- A variable with **global scope** is one whose storage has been created for it by a declaration statement located outside any function.
- A variable with global scope is more commonly termed a **global variable**.
- When using a pointer variable, the value that is obtained is always found by first going to the pointer for an address; this is called **indirect addressing(间接寻址)**.

- To use a stored address, C provides us with an **indirection operator**(间接操作符), *.
- A variable with a **local scope** is simply one that has had storage locations set aside for it by a declaration statement made within a function body.
- Variables created inside a function are available only to the function itself; they are said to be local to the function, or **local variables**.
- A function can invoke a second function, which in turn invokes the first function; this type of recursion is referred to as **indirect** or **mutual recursion**.
- Passing an address is referred to as a function **pass by reference**, because the called function can reference, or access, the variable using the passed address.
- In **pass by value**, a called function receives values from its calling function, stores the passed values in its own local parameters, manipulates these parameters appropriately, and directly returns, at most, a single value.
- A variable that can store an address is known as a **pointer variable**.
- Pointer variables are also **pointers**.
- **Registers(寄存器)** are high-speed storage areas physically located in the computer's processing unit.
- In **run-time initialization(运行时初始化)**, initialization occurs each time the declaration statement is encountered.
- **Scope** is defined as the section of the program where the variable is valid or "known."
- Functions that call themselves are referred to as **self-referential** or **recursive** functions.
- C allocates new memory locations for all function arguments and local variables as each function is called. This allocation is made dynamically, as a program is executed, in a memory area referred to as the **stack**.
- Where and how long a variable's storage locations are kept before they are released can be determined by the **storage class** of the variable. (变量的存储类型决定了变量在释放之前的存储位置和存储时间)

Chapter 8 - Arrays(数组)

Quick Quiz 1

1. What is an atomic variable?

An atomic variable, which is also referred to as a scalar variable, is a variable whose value cannot be further subdivided or separated into a built-in data type. (不能分离称内置数据类型的变量)

2. What is a data structure?

A data structure, which is also known as an aggregate data type, is a data type with two main characteristics. First, its values can be decomposed into individual data elements, each of which is either atomic or another data structure. Second, it provides an access scheme for locating individual data elements within the data structure.(与上述的原子变量相对，由多个数据类型组合而成)

3. A(n) **one-dimensional/single-dimensional/single-subscript(单下标)** array is a list of values of the same data type that is stored using a single group name.
4. Each item in an array is called a(n) **element** or component of the array.

Quick Quiz 2

1. The NULL character (**'\0'**) is automatically appended to all strings by the C compiler.
2. The individual elements of all global and static arrays (local or global) are, by default, set to **zero/0** at compilation time.
3. Are `auto` local arrays initialized at compilation time? If so, to which value?
No. The values within `auto` local arrays are undefined.
4. True/False: It is generally advisable to omit the size of the array in the function header line. **True(建议在函数头省略数组的大小)**

Quick Quiz 3

1. What is a two-dimensional array?
A two-dimensional array, or table(表), consists of both rows and columns of elements.
2. When initializing two-dimensional arrays, the **inner** braces can be omitted.(`int arr[][10]`是可行的)
3. Initialization in a two-dimensional array is done in **row** order.
4. How can you view or interpret arrays of three, four, five, six or more dimensions?
Arrays of three, four, five, six, or more dimensions can be viewed as mathematical *n*-tuples.(*n* 元组)

Key Terms

- One of the simplest data structures, called an **array**, is used to store and process a set of values, all of the same data type that forms a logical group.
- An **atomic variable**, which is also referred to as a **scalar variable(标量变量)**, is a variable whose value cannot be further subdivided or separated into a built-in data type.
- C does not check the value of the index being used (called a **bounds check 边界检查**).
- In **bubble sort**, successive values in the list are compared, beginning with the first two elements.
- A **data structure**, which is also known as an **aggregate data type(聚合数据类型)**, is a data type with two main characteristics. First, its values can be decomposed into individual data elements, each of which is either atomic or another data structure. Secondly, it provides an access scheme for locating individual data elements within the data structure.
- Each item in an array is called an **element** or **component(组件?)** of the array.
- **External sorts(外部排序)** are used for much larger data sets that are stored in large external disk or tape files, and cannot be accommodated within the computer's

memory as a complete unit.(指的是文件体积过大无法一次性载入内存, 需要在外置存储器和内存之间多次交换才能进行排序)

- Each individual element is referred to as an **indexed variable**(下标变量) or a **subscripted variable** because both a variable name and an index or subscript value must be used to reference the element.
- **Internal sorts**(内部排序) are used when the data list is not too large and the complete list can be stored within the computer's memory, usually in an array.
- The two most common methods of performing such searches are the **linear** and **binary search algorithms**.
- In 1958, John McCarthy developed a language at the MIT specifically for manipulating lists; this language was named **LISP**(专门为处理列表的一门语言), the acronym(缩写) for **List Processing**.
- The **NULL** character ('\\0') is automatically appended to all strings by the C compiler.
- A **one-dimensional array**, which is also known as both a **single-dimensional array** and a **single-subscript array**, is a list of values of the same data type that is stored using a single group name.
- The **Quicksort algorithm**, which is also called a "**partition**" sort, divides a list into two smaller sublists and sorts each sublist by portioning into smaller sublists, and so on.
- The third subscript in a three-dimensional array is often called the **rank**(三位数组的第三个下标通常称为秩).
- In a **selection sort**, the smallest value is initially selected from the complete list of data and exchanged with the first element in the list.
- In a linear search, which is also known as a **sequential search**(顺序搜索), each item in the list is examined in the order it occurs until the desired item is found or the end of the list is reached.

Chapter 9 - Character Strings(字符串)

Quick Quiz 1

1. What is a string literal(字符串字面量)?
A string literal is any sequence of characters enclosed in double quotes.
2. What other terms(术语) are used to refer to a string literal?
String constant, string value and string.
3. The NULL character is '\\0'.
4. The newline character is '\\n' .

Quick Quiz 2

1. How does `strcpy(str1, str2)` work?
It copies str2 to str1, including the '\\0'.
2. How does `int toupper(char)` work?

It returns the uppercase equivalent if the character is lowercase; otherwise, it returns the character unchanged.

3. `isalpha()` is included in the **`ctype.h`** header file.
4. `atoi()` is included in the **`stdlib.h`** header file.

Quick Quiz 3

1. What is the difference between the angle brackets and the double quotes in a `#include` statement?
The angle brackets, `<>`, tell the compiler to begin searching for the included file in the C compiler system library directory, while the double quotes, `"`, tell the compiler to start looking in the default directory where the program file is located.
2. The statement `printf("|%25s|", "Have a Happy Day");` displays the message Have a Happy Day, **right**-justified, in a field of 25 characters.
3. What is the result of using the statement `printf("|%-25.12s|", "Have a Happy Day");`?
The statement `printf("|%-25.12s|", "Have a Happy Day");` causes 12 characters to be left justified in a field of 25 characters.(只打出 Have a Happy)
4. True/False: When you use any of the four functions, `printf()`, `scanf()`, `sprintf()`, or `sscanf()`, the control string containing the conversion control sequences need not be explicitly contained within the function.**True**

Key Terms

- A string literal is also referred to as a **string constant** and **string value**, and more conventionally as a **string**.
- A **string literal** is any sequence of characters enclosed in double quotes.

Chapter 10 - Data Files(文件操作、文件流、文件类型)

Quick Quiz 1

1. What is a file?
A file is a collection of data that is stored together under a common name, usually on a disk, magnetic tape, or CD-ROM.
2. What is a file stream?
A file stream is a one-way transmission path that is used to connect a file stored on a physical device, such as a disk or CD-ROM, to a program.
3. Each file stream name, when it is declared, is preceded by a(n) asterisk (*) (星号).
4. If a file opened for reading does not exist, the `fopen()` function returns the **NULL** address value.

Quick Quiz 2

1. The **`void rewind(FILE *fp)`** function resets the current position to the start of the file.

2. What is the role of the `fseek()` function?

The `fseek()` function allows the programmer to move to any position in the file.

3. The function prototype for `ftell()` is contained in `stdio.h`.

4. What happens if a file is opened for output and the file already exists?

When opening a file for output, if the name of an existing data file is used with `fopen()`, the file will be destroyed when it is opened in write mode. (如果以 `w` 模式打开就会把文件销毁)

Quick Quiz 3

1. What are binary files?

Files that are referred to as binary files store numerical values using the computer's internal numerical code. (使用计算机内部的字节码存储)

2. What is a disadvantage of using binary files (instead of text files)?

A disadvantage is that the file can no longer be inspected using either a word processing or text editing program, which means that the ability to see the numerical values as textual information is lost. (失去了以文本形式查看文件的能力)

3. The specification for explicitly creating and writing to a binary file is made by appending a(n) **'b'** to the mode indicator when the file is opened. When using `fwrite()` to write to a binary file, the first method argument is always the **address/**`&` operator and a variable name.

Key Terms

- **Binary files** use the same code as your computer processor uses internally for C's primitive data types.
- Text files are also known as **character-based files**.
- Data that is stored together under a common name on a storage medium other than the computer's main memory is called a **data file**.
- Each file has a unique filename referred to as the file's **external name** (外部名称——不是拓展名: **extended name**).
- A **file** is a collection of data that is stored together under a common name, usually on a disk, magnetic tape, or CD-ROM.
- A **holiday table** consists of legal holiday dates that have been previously stored in a file.
- A file stream that receives (that is, reads) data from a file into a program is referred to as an **input file stream**.
- A file stream that sends (that is, writes) data to a file is referred to as an **output file stream**.
- **Text files** store each individual character, such as a letter, digit, dollar sign, decimal point and so on, using an individual character code (typically ASCII).

Chapter 11 - Arrays, Addresses, and Pointers(数组、地址和指针)

Quick Quiz 1

1. Pointers, both as variables and function parameters, are used to store **addresses**.
2. With respect to pointers, what is an offset(偏移量)?
One unique feature of pointers is that offsets may be included in expressions using pointers. For example, the 1 in the expression `(gPtr + 1)` is an offset. The offset is the number of variables to skip over.*
3. When an array is created, the compiler automatically creates an internal pointer **constant**. (eg: `int arr[5] == int * const arr = (int*)malloc(sizeof(int)*5)`)
4. Can a pointer access be replaced using subscript notation? If so, under which circumstances?(可以将指针加偏移量的操作变为数组下标操作)
A pointer access can always be replaced using subscript notation. For example, if `numPtr` is declared as a pointer variable, the expression `(numPtr + i)` can also be written as `numPtr[i]`. This is true even though `numPtr` is not created as an array.*

Quick Quiz 2

1. What is the purpose of adding or subtracting numbers from pointers?
A pointer, constructed either as a variable or function parameter, contains a value. With pointers, however, the stored value is an address. Thus, by adding numbers to and subtracting numbers from pointers, we can obtain different addresses.
2. When adding or subtracting numbers to pointers, the computer automatically adjusts the number to ensure that the result still “points to” a value of the original **data type**.(自动根据数据类型的进行偏移)
3. How are pointer operations scaled automatically?
*When numbers are added to pointers, a correct scaling is automatically accomplished because the compiler converts the arithmetic operation `pointer + number` to `pointer + number * sizeof(data type being pointed to)`.(编译器自动优化偏移量)*
4. When initializing pointers you must be careful to set a(n) **address** in the pointer.

Quick Quiz 3

1. If `nums` is a two-dimensional integer array, `*(*(nums + 1) + 2)` refers to element **`nums[1][2]`**.
2. What is the main difference between the following declarations?

```
char message1[81] = "this is a string";  
char *message2 = "this is a string";
```

The main difference in the definitions of `message1` as an array and `message2` as a pointer is the way the pointer is created. Defining `message1` using the declaration `static char message1[81]` explicitly calls for a fixed amount of storage for the

*array. This causes the compiler to create a pointer constant. Defining message2 using the declaration `char *message2` explicitly creates a pointer variable first. This pointer is then used to hold the address of a string when the string is actually specified. This difference in definitions has both storage and programming consequences*

3. The header line `int (*calc) ()` declares `calc` to be a pointer to a function that returns an integer.

4. What does the declaration `char *seasons[4];` create?

*The declaration `char *seasons[4];` creates an array of four elements, where each element is a pointer to a character.*

Key Terms

- **Anagram**(变位词) is a rearrangement of the letters in a word or phrase that takes another word or phrase.
- One unique feature of pointers is that **offsets** may be included in expressions using pointers.
- A word, phrase, or sentence that reads the same forward and backward, such as *top spot* is a **palindrome**(回文).
- When an array is created, the compiler automatically creates an internal **pointer constant**(数组名是指针常量) for it and stores the base address of the array in this pointer.

Chapter 12 - Structures(结构体、联合体)

Quick Quiz 1

1. What is the difference between a structure's form and the structure's contents?

A structure's form consists of the symbolic names, data types, and arrangement of individual data fields in the record. The structure's contents consist of the actual data stored in the symbolic names.

2. Assigning actual data values to the data items of a structure is called **populating**(填充) the structure.

3. What is the difference between a homogeneous(同类的) and a heterogeneous(异构的/多项的) data structure?

The difference is best explained with an example. An array is a homogeneous data structure, which means that each of its components must be of the same type. A record is a heterogeneous data structure, which means that each of its components can be of different data types.

4. For non-ANSI C compilers, the keyword **static** must be placed before the keyword `struct` for initialization within a local declaration statement.

Quick Quiz 2

1. A **typedef** statement provides a simple method for creating a new and typically shorter name for an existing structure type.
2. What are parallel arrays?
Parallel arrays are two or more arrays, where each array has the same number of elements and the elements in each array are directly related by their position in the arrays.
3. What is the problem with parallel arrays?
The problem with parallel arrays is that the correspondence between data items is easily lost if only one of the arrays is reordered.
4. When initializing an array of structures, the **inner** braces(内部大括号?) are not necessary.

Quick Quiz 3

1. Individual structure members may be passed to a function in the same manner as any **scalar** variable.(对于结构体依然可以值传递)
2. An alternative to passing a copy of a structure is to pass the **address** of the structure.
3. What is a union?
A union is a data type that reserves the same area in memory for two or more variables, each of which can be a different data type.
4. How much memory space does a union reserve?
A union reserves sufficient memory locations to accommodate its largest member's data type. (额外要说明的是, 这个最大的数据类型所占的字节数如果不是联合体中每一个数据类型字节数的公倍数, 那它会拓展至能成为每一个数据类型的公倍数的大小)

Key Terms

- The structure's **contents** consist of the actual data stored in the symbolic names.
- Each of the individual data items in a "structure" (single unit) is an entity by itself that is referred to as a **data field**(数据字段).
- A structure's **form**(形式) consists of the symbolic names, data types, and arrangement of individual data fields in the record.
- A record is a **heterogeneous data structure**, which means that each of its components can be of different data types.(异构型数据结构)
- An array is a **homogeneous data structure**, which means that each of its components must be of the same type.(数组属于同类型数据结构)
- The data items of a structure are called **members of the structure**.
- **Parallel arrays** are two or more arrays, where each array has the same number of elements and the elements in each array are directly related by their position in the arrays.
- Assigning actual data values to the data items of a structure is called **populating the structure**.
- Taken together, all the data fields form a single unit that is referred to as a **record**(结构体?).

- In C, a record is referred to as a **structure**, and we use these terms interchangeably.
- When defining structures, if the form of the structure is not followed by any variable names, the list of structure members must be preceded by a user-selected **structure type name**.
- A **union** is a data type that reserves the same area in memory for two or more variables, each of which can be a different data type.

Chapter 13 - Dynamic Data Structures(动态数据结构: 链表、栈、队列)

Quick Quiz 1

1. What is a linked list(链表)?
A linked list is a set of structures in which each structure contains at least one member whose value is the address of the next logically ordered structure in the list.
2. What is a self-referencing(自引用) structure?
In a linked list, rather than requiring each structure to be physically stored in the proper order, each new structure is physically added either to the end of the existing list, or wherever the computer has free space in its storage area. The structures are “linked” together by including the address of the next structure in the structure immediately preceding it. From a programming standpoint, the current structure being processed contains the address of the next structure, no matter where the next structure is actually stored. Such structures are also known as self-referencing structures.
3. All programming languages that support pointers provide a special pointer value, known as both NULL and **NIL**, which acts as a sentinel or flag to indicate when the last structure has been processed.
4. The expression `t1.nextaddr->name` can, of course, be replaced by the equivalent expression ***(*t1.nextaddr).name***, which explicitly uses the indirection operator.

Quick Quiz 2

1. What functions are available in C for the dynamic allocation and release of memory space?
*C provides the four functions, **malloc()**, **calloc()**, **realloc()** and **free()**, to control the dynamic allocation and release of memory space.*
2. How does `malloc()` work?
*It reserves the number of bytes requested by the argument passed to the function. It returns the address of the first reserved location as an address of a **void** data type, or **NULL** if sufficient memory is not available.*
3. A(n) **stack** is a special type of linked list in which objects can only be added to and removed from the top of the list.

4. The operation of placing a new structure on the top of a stack is called a PUSH, and removing a structure from a stack is called a(n) **POP**.

Quick Quiz 3

1. What is a queue?
Items are removed from a queue in the order in which they were entered. Thus, a queue is a first in, first out (FIFO) structure.
2. What are the names of the operations used to add and remove items to/from a queue?
Placing a new item on top of the queue is formally referred to as enqueueing(入队), and the operation of removing an item from a queue is formally referred to as serving(服务?).
3. In a(n) **dynamically linked list(动态链表)**, elements can be added and removed from anywhere within the list.
4. The operation of adding a new structure to a dynamically linked list is called a(n) **INSERT**.

Key Terms

- **Dynamic memory allocation** makes it unnecessary to reserve a fixed amount of memory for a scalar, array or structure variable in advance.
- Placing a new item on top of the queue is formally referred to as **enqueueing**.
- The **heap(堆区)** consists of unallocated memory that can be allocated to a program as requested, while the program is executing.
- The field on which a list is ordered is referred to as the **key field(关键字段)**, and insertions and deletions are always made to preserve the ordering of this field.
- A **linked list** is a set of structures in which each structure contains at least one member whose value is the address of the next logically ordered structure in the list.
- Items are removed from a **queue** in the order in which they were entered.
- Dynamic memory allocation is also known as **run-time allocation**.
- Structures that are “linked” together by including the address of the next structure in the structure immediately preceding it are known as **self-referencing structures**.
- The operation of removing an item from a queue is formally referred to as **serving**

Chapter 14 - Additional Capabilities(额外特性：条件编译、三元运算符、位运算、枚举)

Quick Quiz 1

1. Both the `#ifndef` and `#ifdef` directives permit **conditional compilation(条件编译)** in that the statements immediately following these directives, up to either the

`#else` or `#endif` directives, are compiled only if the condition is true, whereas the statements following the `#else` are compiled only if the condition is false.

2. Are there any ternary operators in C? If so, give an example of a ternary operator.

The conditional operator, `?:`, is unique in C in that it is a ternary operator(三元运算符).

3. The **`typedef`** statement creates a new name for an existing data type.

4. How can you create enumerated lists(枚举列表) in C?

*Enumerated lists are identified by the reserved word **`enum`** followed by an optional, user-selected name and a required list of one or more constants.*

Quick Quiz 2

1. What are bit operators?

The operators that are used to perform bit operations in C are known as bit operators.

The bit operators in C are `&`, `|`, `^`, `~`, `<<` and `>>`.

2. In an operation like `op1 & op2`, the 0s in `op2` effectively mask, or eliminate, the respective bits in `op1`, while the 1s in `op2` *filter/pass* the respective bits in `op1` through with no change in their values.(`op1` 中的一些位被 `op2` 中的 0&成了 0)
3. In a(n) **arithmetic** right shift (using the `>>` operator), each single shift to the right corresponds to a division by 2.
4. How does the `^` operator work?

The exclusive OR operator, `^`, performs a bit-by-bit comparison of its two operands.

The result of the comparison is determined by the following rule: The result of an exclusive OR comparison is 1 if one and only one of the bits being compared is a 1; otherwise the result is 0. (两个数只能有且只有一个数的这一位为 1)

Quick Quiz 3

1. What is a macro(宏)?

*When the equivalence created using a **`#define`** statement consists of more than a single value, operator or variable, the symbolic name is referred to as a macro, and the substitution of the text in place of the symbolic name is called a macro expansion or macro substitution(宏替换).*

2. What are command-line arguments?

They are arguments that are typed on the command line.

3. The advantage of using a(n) macro instead of a function is an increase in execution speed.
4. Any argument typed on a command line is considered to be a(n) **string**.

Key Terms

- The `typedef` declaration statement permits constructing alternate names for an existing C data type name. These alternate names are known as **aliases**(别名).
- In an **arithmetic right shift** (using the `>>` operator), each single shift to the right corresponds to a division by 2.
- The operators that are used to perform bit operations in C are known as **bit operators**.

- **Command-line arguments** are arguments that are typed on the command line.
- Both the `#ifndef` and `#ifdef` directives permit **conditional compilation** in that the statements immediately following these directives, up to either the `#else` or `#endif` directives, are compiled only if the condition is true, whereas the statements following the `#else` are compiled only if the condition is false.
- In an operation like `op1 & op2`, the 0s in `op2` effectively mask, or eliminate, the respective bits in `op1`, while the ones in `op2` **filter**, or pass, the respective bits in `op1` through with no change in their values.
- For positive signed numbers, where the leftmost bit is 0, both arithmetic and **logical right shifts** produce the same result.(逻辑右移不考虑符号位, 而算数右移需要考虑符号位, 如果最左位为 1, 右移之后最左位需要补 1)
- When the equivalence created using a `#define` statement consists of more than a single value, operator or variable, the symbolic name is referred to as a **macro**, and the substitution of the text in place of the symbolic name is called a **macro expansion** or **macro substitution**.
- In an operation like `op1 & op2`, the variable `op2` is called a **mask**(屏蔽).
- AND operations are extremely useful in **masking**, or eliminating, selected bits from an operand.
- The conditional operator, `?:`, is unique in C in that it is a **ternary operator**.