



现代制造学院 | 香港铸业学院

C 语言





授课主题

授课主题的详细内容

教学目标

教学目标的详细内容

教学重点、难点与解决方案

重点：

难点：

学时分配



顺序结构程序设计

01

C语言的基本语句

02

数据输入与输出

03

程序举例

04

总结





01

PART ONE

C语言的基本语句





程序设计语言有顺序、选择、循环三种基本的控制结构。

顺序结构是最基本的控制结构，其包含的语句是按照书写的顺序执行的，且每条语句都将被执行。程序流程如图 1 所示，语句按书写顺序执行。先执行 A，再执行 B。

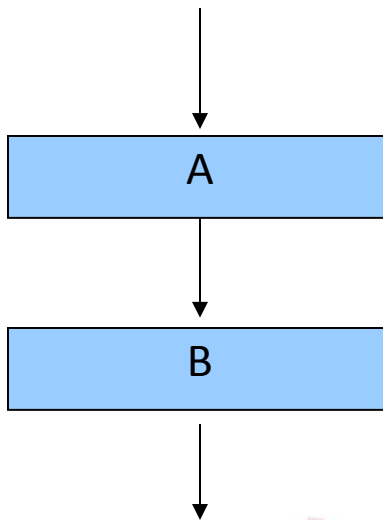


图 1 顺序结构执行流程

例：`#include<stdio.h>`

```
main( )
```

```
{int a=10,x=5,y=6;
```

```
  a+=a*=6;
```

```
  x=y++;
```

```
  y=++x ;
```

```
  a=x+++y;
```

```
  printf(“%d,%d,%d”,a,x,y);
```

```
}
```

输出：

14 , 8 , 7



C 语言的语句可分为以下四类:

简单语句（包括输入 / 输出语句）

空语句

复合语句

流程控制语句。

本章将介绍前三种语句。





简单语句

1. 表达式语句：

由一个表达式加上一个分号构成。

如： $A++$; $x=1$;

$p+=q*4+5$; $y=4>2?6:1$;

2. 空语句：

由一个分号表示，一般形式为：

;

空语句在语法上是一条语句，但执行时
不做任何操作。



3. 函数调用语句：

由函数调用表达式加分号构成。

如：`printf(“%d”,a);` `scanf(“%f”,&b);`

C 语言有丰富的标准函数库，可提供各类函数供用户调用，完成预先设定好的操作。

例如调用标准库函数求数学函数值：`sin(x)`、`cos(x)`、`exp(x)`（求 e^x ）、`fabs(x)`（求 x 的绝对值）、`log(x)` 等。





调用标准库函数要注意

- 在程序中要包含相应的头文件

例如：`#include<stdio.h>`

`#include<math.h>`

- **调用规则：函数名（参数表）**

例如：`y=sin(1.7); x=pow(3,18);`

- **函数调用的实质：**

**在调用点转去执行一段预先设计好的程序，
求出结果后返回调用点。**

函数的值又叫函数的返回值。



**如：数学表达式 $y=|3\sin x+4\cos x|+e^x$ ，
可以用 C 语言表达式表示为：**

```
y=fabs(3*sin(x)+4*cos(x))+exp(x);
```

其中，fabs,sin,cos,exp 都属于头文件 math.h，用到这些函数，必须在程序的预处理语句中用 #include<math.h> 把数学函数库包含进来。





复合语句

用一对花括号，把若干条语句括起来，就形成了一条复合语句。形式如下：

```
{  
    语句 1;  
    .....  
    语句 n;  
}
```

花括号中的语句，可以是简单语句、空语句、复合语句、流程控制语句，所有这些语句括在一起，在语法上看成是一条语句，执行时顺序执行花括号中的每条语句。复合语句多用于流程控制语句中。



02

PART TWO

数据输入与输出





数据输入与输出

C语言本身不提供输入输出语句，得由标准 I/O 库函数提供。输入输出库函数很丰富，可以从键盘、显示器、磁盘文件和硬件端口进行输入或输出操作。

本节主要介绍用于键盘输入和显示器输出的函数。





字符输入输出函数

1. 字符输出函数

`int putchar(int)`

向标准输出设备（一般为显示器）输出一个字符，并返回该字母的 ASCII 码值。参数可以是字符常量、变量或整型常量、变量。

如：`#include<stdio.h>`

`main()`

`{ int a=65; char c='d';`

`putchar(a); putchar(97); putchar('\n');putchar(c);`

`}`

输出为： Aa
d





2. 字符输入函数

int getchar(void)

从输入设备（一般为键盘）上输入一个字符。它每被调用一次，就从标准输入设备上取一个字符，返回值是该字符的 ASCII 编码值，可以赋给字符变量或整型变量。

注意

执行 `getchar()` 时输入字符以回车结束，同时回车字符也可以做为输入的字符。

思考：以下三种不同的输入，输出情况如何？

1. abcdefg

2. ab

3. a

bcdefg

```
例：#include<stdio.h>  
main()  
{ int i, j; char c;  
    i=getchar(); putchar(i);  
    j=getchar(); putchar(j);  
    c=getchar(); putchar(c);  
}
```



格式输出函数

前面两条语句非常方便，但一次只能处理一个字符，

格式输出函数可以按规定格式向输出设备（一般为显示器）输出数据，并返回输出的字符数。这个函数可以输出多字信息。





格式输出函数的一般形式

`printf(“格式控制字符串” , 输出参数表);`

用双引号括起来，
控制输出项的格式
和输出一些提示信息

可以是一个或多个输出项。
可以是常量或变量表达式，
用逗号分隔。类型可以是
整型实型字符型和字符串
型。

具体的讲，`printf` 语句，就是把输出参数表中的表达式的值，按照格式控制字符串的格式，依次在指定的位置输出，如果输出位置不够，从左到右依次输出前几个的值。



格式输出函数运用示例：

1. `printf("It's fun!");`

输出： It's fun!

2. `int a=1,b=2;`

`printf("%d,%d",a,b);`

输出： 1 , 2

3. `int a=1,b=2;`

`printf("a=%d\nb=%d",a,b);`

输出： a=1
b=2

4. `int a=1,b=2;`

`printf("%d",a+b);`

输出： 3

5. `int a=1,b=2;`

`printf("a+b");`

输出： a+b



在 printf 语句的格式控制字符串中，有三类字符：

- **普通字符**：按原样输出，主要起提示作用。
- **转义字符**：指明特定的操作，如 \n 换行、\t 横向跳格
- **格式说明部分**：由 % 引导的格式字符串组成

输出参数表中的表达式，就是按照格式说明部分指定的格式，在格式控制字符串中的相应位置输出的。

格式说明部分的一般格式为：

%[flags][width][.prec][F|N|h|L][type]

在这里，我们对常用的几种形式进行介绍！





这是最简单的格式输出说明形式，说明以何种形式输出表达式的值。

type 字符表

d	以带符号的十进制形式输出整数（正数不输出符号）
o	以 8 进制无符号形式输出整数（不输出前导符 0）
x(X)	以 16 进制无符号形式输出整数（不输出前导符 0x）
u	以无符号 10 进制形式输出整数
c	以字符形式输出一个字符
s	输出字符串
f	以小数形式输出单、双精度数，隐含输出 6 位小数
e(E)	以标准指数形式输出单、双精度数，小数位数为 6 位
g(G)	选用 %f 或 %e 格式中输出宽度较短的一种格式



表达式值的类型

对应

type 格式

字符型 (char 、 unsigned char)

%c 、 %d 、 %u 、 %o 、 %x

整型 (int 、 unsigned)

%d 、 %u 、 %o 、 %x 、 %c

(long 、 unsigned long)

%ld 、 %lu 、 %o 、 %x 、 %c

实型 (float)

%f 、 %e 、 %g

(double)

%lf 、 %e 、 %g

字符串 (字符串常量、字符数组)

%s

在字符型和整型之间，输出的 type 格式是可以互换的，而与实型及字符串型则不能。

在 % 与 type 字符之间加一个字符 l 表示输出长整型或双精度型数据的值。



2. .prec

%f 与 %lf 的输出格式都是在小数点后保留六位数字，有余四舍五入，缺少以 0 补足。

在 C 语言中，输出实数时可以用在 % 与 type 符之间加入 .prec 的格式来控制输出的小数位数。

例： `printf("%.3f , %.2f", 3.1415926 , 3);`

输出为： `3.142 , 3.00`

.prec 也可以加在 % 与整型、字符型 type 符之间，意义各不一样，此处不作介绍。





```
例：  
#include<stdio.h>  
  
main( )  
{ int a=65,i= -2;char c='a';  
  float b=1.34576;double d=4.65;  
  unsigned j=65535;  
  printf("\n%s","hello");  
  printf("\n%d,%c,%d,%u,%u,%d",a,a,i,i,j,j);  
  printf("\n%.3f,%.4lf",b,d);  
  printf("\n%d,%d",a+i,i++);  
}
```

输出：

hello

65,A,-2,65534 ,65535, -1

1.346,4.6500

64,-2





格式输入函数

格式输入函数一般形式

`scanf(“ 格式控制字符串” , 地址列表);`

在 `scanf` 语句的格式控制字符串中，有两类字符：

- 普通字符：按原样输入
- 格式说明部分：由 % 引导的格式字符串组成

`scanf` 语句地址列表中的地址，用取址符 & 加变量名的形式表示，指的是该变量所代表的内存空间的地址。

`scanf` 语句在运行时，会停下来，等待从键盘输入值依次存入地址列表中的地址空间，即相当于从键盘给变量赋值。





输入数据时要严格按照 scanf 语句中格式控制字符串的形式输入。看下面的例子：

1. scanf(“%d%d”,&a,&b);

输入：1 2 回车 或者 1 回车 2 回车

2. scanf(“%d,%d”,&a,&b);

输入：1,2 回车

3. Scanf(“a=%d,b=%d”,&a,&b);

输入：a=1,b=2 回车





scanf 语句格式说明部分的一般格式为：

`%[*][width][F|N][h|L]type`

`type` 是必须的，表示输入后转换的数据类型。

<code>d(D)</code>	十进制整数
<code>o(O)</code>	八进制整数
<code>x(X)</code>	十六进制整数
<code>i(I)</code>	自动根据数头分辨十、八、十六进制
<code>u(U)</code>	无符号十进制整数
<code>c</code>	单个字符
<code>s</code>	字符串
<code>f(e,g,G)</code>	实数
<code>n</code>	不输入数据，将已读入的字符输送到对应的地址中
<code>%</code>	百分号

type 字符表



3. 使用 scanf 应注意：

- ✓ **scanf () 的格式控制字符串中的字符不是用于输出的，都是要求输入的。 scanf (" x=%d" , &x) ;**
- ✓ **参数的第二部分一定是地址列表，不能是表达式，请记得在变两前加上 &。**
- ✓ **执行 scanf () 输入数据时，在两个数据之间允许以一个或多个空格间隔，也可以用回车键、 tab 键分隔。**
- ✓ **实数不许规定精度，像 %10.4f 是不合法的。**
- ✓ **%后面有 * 号时，该数据会被禁止使用。**
- ✓ **%后面有数字，可以自动截取该长度的数据作为输入值。**
- ✓ **如果输入时类型不匹配则停止处理，返回 0。**



03

PART THREE

程序举例





程序举例

例 1、输入圆的半径，输出圆的周长和面积。

分析：

1. 定义实型变量 r 、 l 、 s 用于存放半径、周长、面积；
2. 调用输入函数，输入 r ；
3. 分别利用周长公式和面积公式求出 l 、 s ；
4. 调用输出函数输出 l 、 s 。





程序：

```
#include <stdio.h>

main()

{ float r , l , s ;

  scanf(“ %f ” , &r) ;

  l = 2 * 3.14159 * r ;

  s = 3.14159 *r * r ;

  printf(“\n l=%.4f” , l) ;

  printf(“\n s=%.4f” , s) ;

}
```





例 2 从键盘输入一个小写字母，用大写形式输出该字母。

分析：

- 1. 输入小写字母存入变量 a ；**
- 2. 转换成大写 $a = a - 32$ ；**
- 3. 输出 a .**





程序

```
#include "stdio.h"
```

```
void main()
```

```
{ char a ;  
  printf("Input a lowercase letter : " ) ;  
  a = getchar() ;  
  a = a-32 ;  
  printf("%c \n" , a) ;
```

思考：如何使程序可以处理任何字符的输入。即判断输入的是小写字母，才转换成大写；如果输入的其他字符，则直接输出。

```
}
```

例求 $ax^2 + bx + c = 0$ 方程的根，设 $b^2 - 4ac > 0$

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

```
/*ch3_14.c*/
#include <stdio.h>
#include <math.h>
main()
{ float a,b,c,disc,x1,x2,p,q;
  scanf("a=%f,b=%f,c=%f",&a,&b,&c);
  disc=b*b-4*a*c;
  p=-b/(2*a); q=sqrt(disc)/(2*a);
  x1=p+q; x2=p-q;
  printf("\n\nx1=%5.2f\nx2=%5.2f\n",x1,x2);
}
```

输入 : a=1,b=3,c=2 ↵

输出 : x1=-1.00

x2=-2.00



04

PART FOUR

总结





SOC 达成度的总体评价

效果

教学反思与小结

效果





广东岭南职业技术学院
GUANGDONG LINGNAN
POLYTECHNIC

感谢各位聆听

Thanks for Listening



现代制造学院 | 香港铸业学院

模具设计与制造专业

