



石家莊鐵道大學  
SHIJIAZHUANG TIEDAO UNIVERSITY

在线开放课程

C语言程序设计

—简单的C程序设计

数据的输入输出

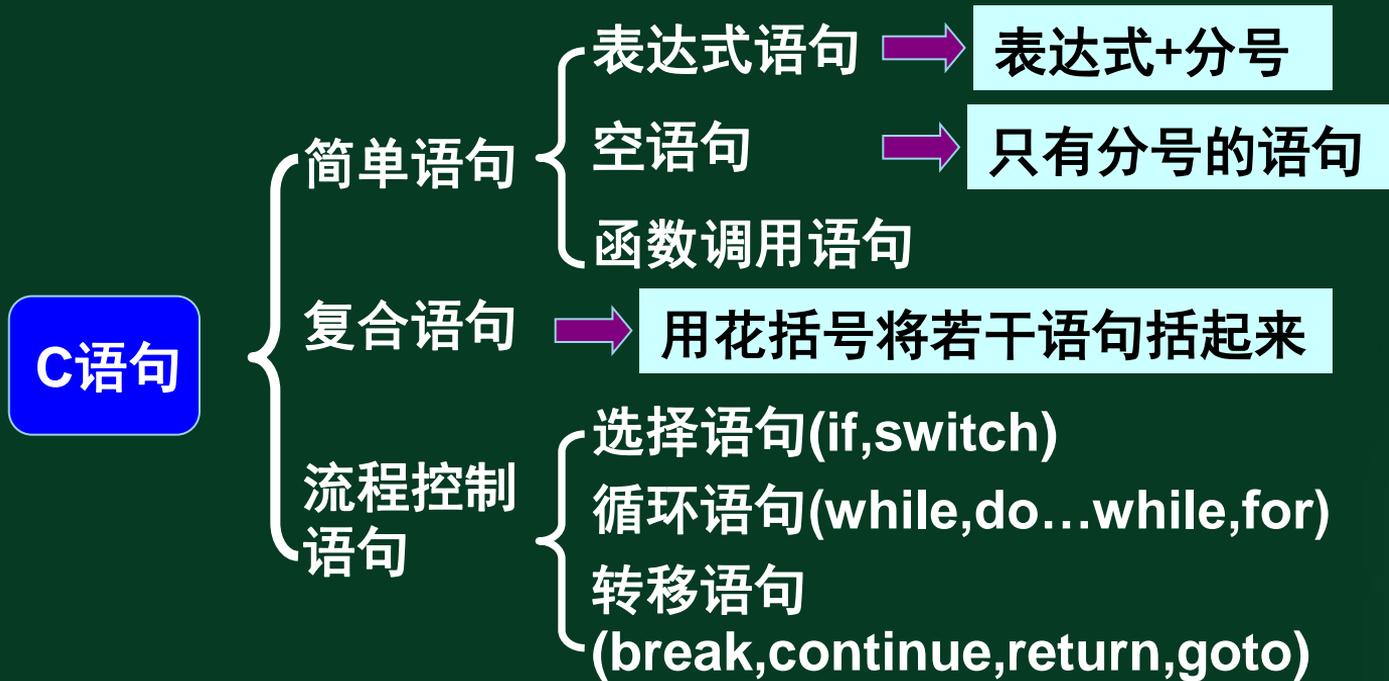
主讲：马银华

# 目录

---

- ◆ 1、C语言的语句
- ◆ 2、赋值语句
- ◆ 3、数据的输入输出
- ◆ 4、顺序结构程序举例

# 一、C语言的语句



**注意：**复合语句的“}”后面不能出现分号，而“{”前复合语句中最后一条语句的分号不能省略。

## 二、赋值语句

### 1 赋值语句—赋值语句的一般形式

变量 = 表达式 ;

赋值语句与赋值表达式不一样，赋值语句可以作为程序中一个独立的程序行，而赋值表达式是作为一个运算量，可以出现在表达式中。

## 二、赋值语句

### 1 赋值语句—课堂练习

若变量均已正确定义并赋值，以下合法的C语言赋值语句是（ ）。

A.  $x=y=5$  ;

B.  $x=n\%2.5$ ;

C.  $x+n=l$  ;

D.  $x=5=4+1$ ;

## 二、赋值语句

```
#include <stdio.h>
#include <math.h>
int main ( )
{ double a,b,c,s,area; // 定义各变量, 均为double型
  a=3.67; // 对边长a赋值
  b=5.43; // 对边长b赋值
  c=6.21; // 对边长c赋值
  s=(a+b+c)/2; // 计算s
  area=sqrt(s*(s-a)*(s-b)*(s-c)); // 计算area
  printf("a=%f\tb=%f\t%f\n",a,b,c); // 输出三边a,b,c的值
  printf("area=%f\n",area); // 输出面积area的值
  return 0;
}
```

功能描述：给出三角形的三个边长，求三角形面积

## 二、赋值语句

### 3 赋值时数据类型的转换

当赋值运算符两边的数据类型不一致时，C编译系统自动将赋值运算符右边表达式的数据类型转换成左边变量相同的类型

转换规则：

- ①整型变量与实型变量之间的转换
- ②不同精度实型变量之间的转换
- ③不同精度整型变量之间的转换

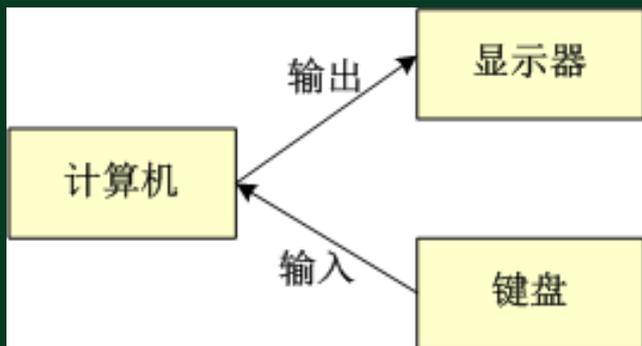
截取有效数字

截取低字节数据

# 三、数据的输入输出

## 1 数据输入输出的引入

输入输出以计算机为主体而言



C语言本身不提供输入输出语句,需要**输入输出函数**完成。

`stdio.h`

# 三、数据的输入输出

## 1 数据输入输出的引入

为了实现输入和输出功能，在C的库函数中提供了一组输入输出函数：

```
getchar();  
putchar(字符变量);  
scanf(“格式控制字符串”，地址表列);  
printf(“格式控制字符串”，输出表列);
```

在使用C语言库函数时，要用预编译命令**#include**将有关“**头文件**”包括到源文件中

```
#include <stdio.h>
```

```
#include “stdio.h”
```

# 三、数据的输入输出

## 2 字符输入输出函数

putchar函数（字符输出函数）

向终端输出  
一个字符

一般形式 `int putchar(字符变量)`

getchar函数（字符输入函数）

从键盘上输  
入一个字符

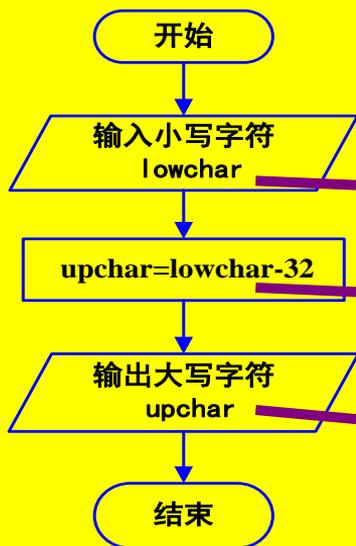
一般形式 `int getchar()`

getchar( )

putchar(字符变量)

## 2 字符输入输出函数—示例

引例：从键盘输入一个，  
输出对应的大写字母



```
#include <stdio.h>
main()
{
    char lowchar,upchar;
    lowchar=getchar();
    upchar=lowchar-32;
    putchar(upchar);
}
```

# 三、数据的输入输出

## 3 格式化输出函数

一般形式：

```
printf(“格式控制字符串”, 输出表列);
```

指定输出格式，由**格式字符串**和**非格式字符串**两种组成，**非格式字符串**照原样输出。

# 三、数据的输入输出

## 3 格式化输出函数

格式字符	意义
d,i	以十进制形式输出带符号整数(正数不输出符号)
o	以八进制形式输出无符号整数(不输出前缀0)
x,X	以十六进制形式输出无符号整数(不输出前缀0x)
u	以十进制形式输出无符号整数
f	以小数形式输出单、双精度实数
e,E	以指数形式输出单、双精度实数
g,G	以%f或%e中较短的输出宽度输出单、双精度实数
c	输出单个字符
s	输出字符串

# 三、数据的输入输出

## 3 格式化输出函数

一般形式：

**%**[标志][输出最小宽度][.精度][长度]**类型**

标志	意 义
-	结果左对齐，右边填充空格
+	输出符号(正号或负号)
空格	输出值为正时冠以空格，为负时冠以负号
#	对c,s,d,u类无影响；对o类,在输出时加前缀o；对x类,在输出时加前缀0x

# 三、数据的输入输出

## 3 格式化输出函数

一般形式：

**%**[标志][输出最小宽度][.精度][长度]**类型**

■ 输出最小宽度：用十进制整数来表示。

■ 精度：若实际位数多于定义的宽度，则按实际位数输出；若实际位数小于定义的

■ 长度：如果输出数字，则表示小数的位数；如果输出的是字符，则表示输出字符的个数；若实际位数大于所定义的精度数，则截去超过的部分。

# 三、数据的输入输出

## 3 格式化输出函数—示例

例2：用%f输出实数，输出结果显示6位小数

```
#include <stdio.h>
int main()
{
    double a=1.0;
    printf("%f\n",a/3);
    return 0;
}
```

```
0.333333
Press any key to continue
```

## 数据的输入输出

**%[标志][输出最小宽度][.精度][长度]类型**

### 3 格式化输出函数—示例

#### 例3 分析下面程序的运行结果

```
main()
{
    unsigned int a=65535;
    int b=-2;
    printf("a=%d,%o,%x,%u\n",a,a,a,a);
    printf("b=%d,%o,%x,%u\n",b,b,b,b);
}
```

输出(Turbo 2.0):

a=-1,177777,ffff,65535  
b=-2,177776,fffe,65534

输出(VC++6.0):

a=65535,177777,ffff,65535  
b=-2,377777777776,ffffffe,4294967294

## 数据的输入输出

**%[标志][输出最小宽度][.精度][长度]类型**

### 3 格式化输出函数—示例

#### 例4 分析下面程序的运行结果

```
main()
{char c='a';
 int i=97;
 printf("%c,%d\n",c,c);
 printf("%c,%d\n",i,i);
}
```

输出:

a, 97

a, 97

# 三、数据的输入输出

## 3 格式化输出函数—示例

### 例5 分析下面程序的运行结果

```
main()
{
    float x=1234.789012f;
    double y=123456789012.123456;
    printf("x=%f,y=%f\n",x,y);
}
```

**float**型有**7位**有效数字

**double**型有**16位**有效数字

输出结果：

**1234.789063,y=123456789012.123460**

# 三、数据的输入输出

## 3 格式化输出函数—示例

### 例6 分析下面程序的运行结果

```
#include <stdio.h>
main()
{ float f=123.456;
  printf("%f,%10.2f,%10.4f\n",f,f,f);
  printf("%e,%10.2e,%10.4e\n",f,f,f);
}
```

输出结果：

```
123.456001,      123.46,      123.4560
1.234560e+002, 1.23e+002,1.2346e+002
```

# 三、数据的输入输出

## 4 格式化输入函数

一般形式：

**scanf (“格式控制字符串”， 地址表列);**

指定输入格式， 只有格式字符串。

# 三、数据的输入输出

## 4 格式化输入函数

格式	字符意义
d, i	输入十进制整数
o	输入八进制整数
x	输入十六进制整数
u	输入无符号十进制整数
f或e	输入实型数(用小数形式或指数形式)
c	输入单个字符
s	输入字符串

# 三、数据的输入输出

## 4 格式化输入函数

**%[\*][输入数据宽度][长度]类型**

- “\*” 符：用以表示该输入项,读入后不赋予相应的变量,即跳过该输入值。
- 宽度：用十进制整数指定输入的宽度(即字符数)。
- 长度：长度格式符为l(L)和h, l表示输入长整型数据(如%ld)和双精度浮点数(如%lf)。h表示输入短整型数据。

# 三、数据的输入输出

## 4 格式化输入函数—练习

用格式化输入方式修改引例：从键盘输入一个小写字母，要求在屏幕上输出对应的大写字母

```
#include <stdio.h>
main()
{
    char lowchar,upchar;
    scanf("%c",&lowchar);
    upchar=lowchar-32;
    printf("%c\n",upchar);
}
```

# 三、数据的输入输出

## 4 格式化输入函数—几点说明

①scanf函数中要求给出变量地址，而不是变量名。

```
scanf("%d",a);
```

✗

```
scanf("%d",&a);
```

✓

②在输入字符数据时，若格式控制串中无非格式字符，则认为所有输入的字符均为有效字符。

```
scanf("%c%c%c",&c1,&c2,&c3);
```

如输入 a\_b\_c

则把 'a' 赋予c1, ' ' 赋予c2, 'b' 赋予c3

## 四、顺序结构程序举例

例7： 输入整数a和b，交换a和b后输出。

解题思路：  
如何完成交换过程？

```
temp=a;
```

```
a=b;
```

```
b=temp;
```

15

a

36

b

15

36

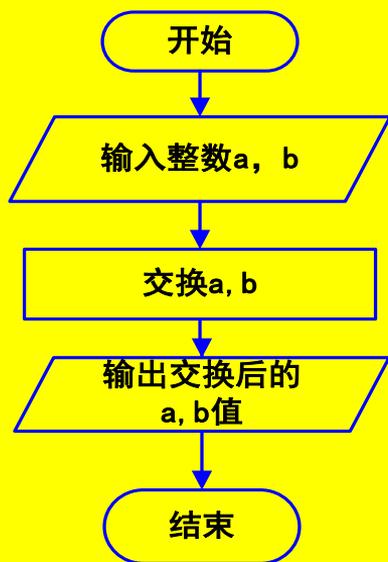
temp



# 四、顺序结构程序举例

例7： 输入整数a和b，交换a和b后输出。

编程思路流程图：



编写程序代码

## 四、顺序结构程序举例

例8：从输入圆的半径，输出圆的周长和面积。

```
#include <stdio.h>
#define PI 3.1415
main()
{   float Radius;
    float Circumference,Area;
    scanf("%f",&Radius);
    Circumference=2*(float)PI*Radius;
    Area=(float)PI*Radius*Radius;
    printf("Circumference=%f,Area=%f\n",
           Circumference,Area);
}
```

- ◆ 接下来将学习
- ◆ 数据的输入与输出

# 谢谢！