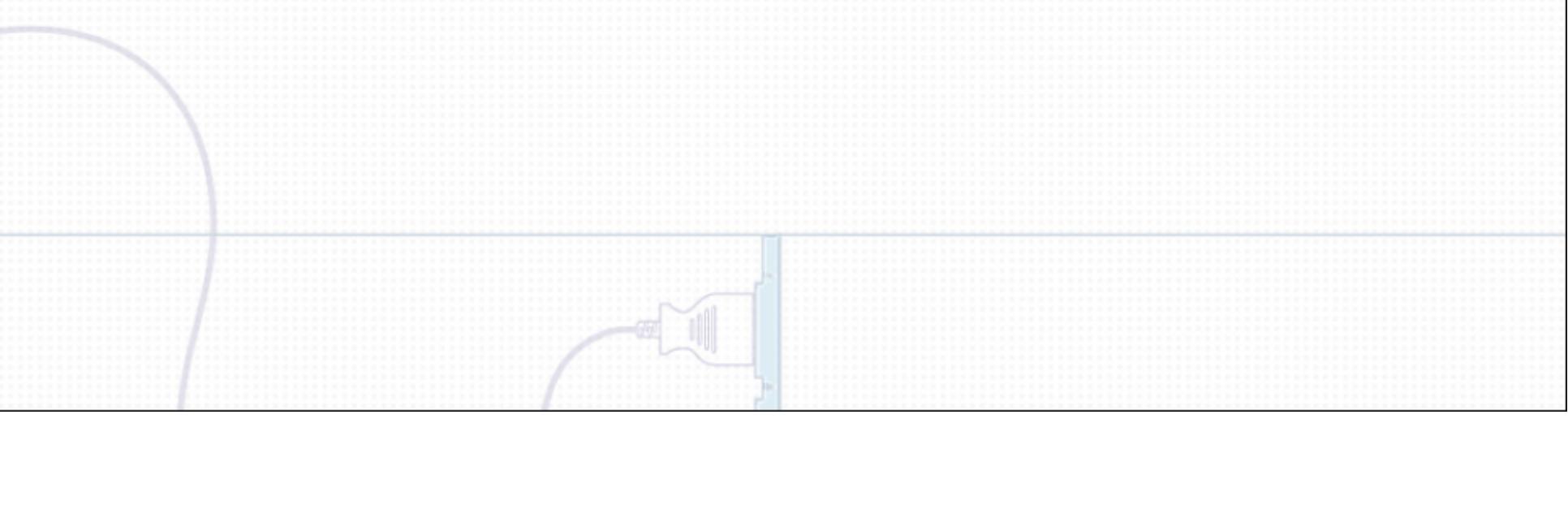


## 项目2

# 从鸡兔同笼问题开始-C语言的基础



# 学习目标

## ●知识目标

- (1) 掌握常量和变量的概念及使用。
- (2) 掌握C语言的基本数据类型。
- (3) 掌握各种数学运算符的使用方法。
- (4) 掌握数学表达式转换成C语言的表达式

## ●能力目标

- (1) 能够定义和使用变量。
- (2) 能够通过赋值语句为变量赋值和输出变量的值。
- (3) 能够根据运算符的优先级和结合性计算表达式的值。
- (4) 能够应用熟练使用Dev-C++进行C程序的编辑、编译和执行



# 任务1 常量与变量

## 【任务导入】

我国有个非常有名的古典数学题：鸡兔同笼问题。问题的题目是这样的：有若干只鸡兔同在一个笼子里，从上面数，有16个头；从下面数，有40只脚。问笼中各有几只鸡和兔？

从数学上解这个题目非常容易。解法如下：

设：鸡数量为 $x$ ，兔数量为 $y$ ，总头数为 $h=16$ ，  
总脚数为 $f=40$ 。列出二个方程：

$$\begin{cases} x+y=h & (1) \\ 2x+4y=f \end{cases}$$

解得：

(2)

$$y = \frac{f - 2h}{2}$$

$$x = y - h$$

# 任务1 常量与变量

## 一、C语言的的字符集和词汇



### 1、C语言字符集

字符是组成语言的最基本的元素，C语言字符集由字母、数字、空格、标点和特殊字符组成。

- 1) .字母 小写字母a-z共26个。 大写字母A-Z共26个
- 2) .数字 0~9共10个
- 3) .空白符：是空格符、制表符、换行符的统称
- 4) .下划线、标点和运算字符：

# 任务1 常量与变量

## 一、C语言的的字符集和词汇



表1-1 C语言标点和运算字符

字符	名称	字符	名称	字符	名称	字符	名称
,	逗号	( )	圆括号	\	反斜杠	/	除号
.	圆点	[ ]	方括号	~	波折号	+	加号
;	分号	{ }	花括号	#	井号	-	减号
?	问号	< >	尖括号	%	百分号	=	等号
'	单引号	>	大于号	&	与		竖线
"	双引号	<	小于号	^	异或	_	下划线
:	冒号	!	惊叹号	*	乘号		

# 任务1 常量与变量



## 一、C语言的的字符集和词汇

### 2、C语言的词汇

就象英文字母构成英文单词一样，上述的字符集组成了C语言的词汇。在C语言中使用的词汇分为六类：标识符，关键字，运算符，分隔符，常量，注释符等。

#### 1) 标识符

在程序中凡是用来标记变量名、函数名、文件名、标号的字符序列称为标识符。

C语言规定，标识符由首部和其他部分构成。标识符只能是字母数字、下划线组成的字符串。首部（姓）必须是字母或下划线，其他(名)没有要求。另外大小写字母是不同的字符。

# 任务1 常量与变量

## 一、C语言的的字符集和词汇



### 2) 关键字

关键字就是C语言系统自己保留的标识符。所有的关键字都有固定的含义，不能用做其他用途，只能小写。

### 3) 运算符

C语言中含有相当丰富的运算符。运算符、变量与函数一起组成表达式，表示各种运算功能。

### 4) 分隔符

在C语言中采用的分隔符有逗号和空格两种。

### 5) 注释符

### 6) 常量

# 任务1 常量与变量

## 二、数据的变与不变 — 变量、常量

在C语言中把计算中不变的量叫做“**常量**”。

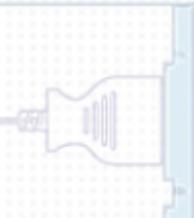
比如：鸡兔同笼的问题中总头数16和总脚数40

圆周率3.14159265

把计算中变化的这些量叫做“**变量**”，

比如：鸡和兔的只数是随给出的头数和脚数而变的。

计算圆的面积时圆的半径



# 任务1 常量与变量

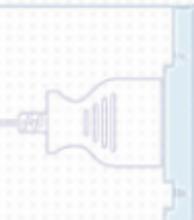
## 三、整型常量

不同数制整型常量的表示

例如: 12 : 十进制数 12

012 : 八进制数 12 (等于十进制数10,用前导符0表示八进制数常量)

0x12: 十六进制数 12 (等于十进制数 18, 用前导符0X或0x表示十六进制数常量)



# 任务1 常量与变量

## 四、整型变量的定义及赋值

变量就是类似于解方程时的设未知数。

变量代表内存中具有特定属性的一个存储单元，它用来存放数据，这就是变量的值（相当未知数的解）。

变量也要有名字。命名规则符合前面标识符的命名规则。

**在C语言中规定所有的变量必须先定义后使用。**

整型常量的表示:最常用的就用int来定义。就是Integer的简写。

```
int chicken;
```

```
int rabbit;
```

```
int head;
```

```
int foot;
```



# 任务1 常量与变量

## 四、整型变量的定义及赋值

整型常量的赋值：

```
int head=16;
```

```
int foot=40;
```

注意：数学中 $head=16$ 和 $16=head$ 是一回事。但在C语言中却不能反过来写。

“=” 是赋值符 (Assign)



# 任务1 常量与变量

## 五、数学表达式的转换

$$y = \frac{f - 2h}{2}$$

必须转换为： $y = (f - 2 * h) / 2$  。

**注意：**数学上 $2h$ 和 $2 \times h$ 是一回事，但是在C语言中 $2h$  必须用 $2 * h$ 来表示。

$$x = y - h$$

另一个式子 无需转换



# 任务1 常量与变量

## 六、在屏幕上显示整数—输出

用 `printf ( " %d\n " ,rabbit);`

不能用 `printf ( “ rabbit \n ” );`



# 任务1 常量与变量

## 【任务实施】

鸡兔同笼问题程序代码：

```
#include <stdio.h>

int main ()

{ int chicken,rabbit,head,foot;

  head=16;

  foot=40;

  rabbit=(foot-2*head)/2;

  chicken= rabbit-head;

  printf( " %d \n " , rabbit) ;

  return 0;

}
```



# 任务2 基本数据类型

## 一、C语言基本数据类型

一个程序应包括两个方面的内容：

- 对数据的描述：数据结构(data structure)
- 对操作的描述：算法(algorithm)

著名计算机科学家沃思提出一个公式

:



**数据结构 + 算法 = 程序**



完整的程序设计应该是：

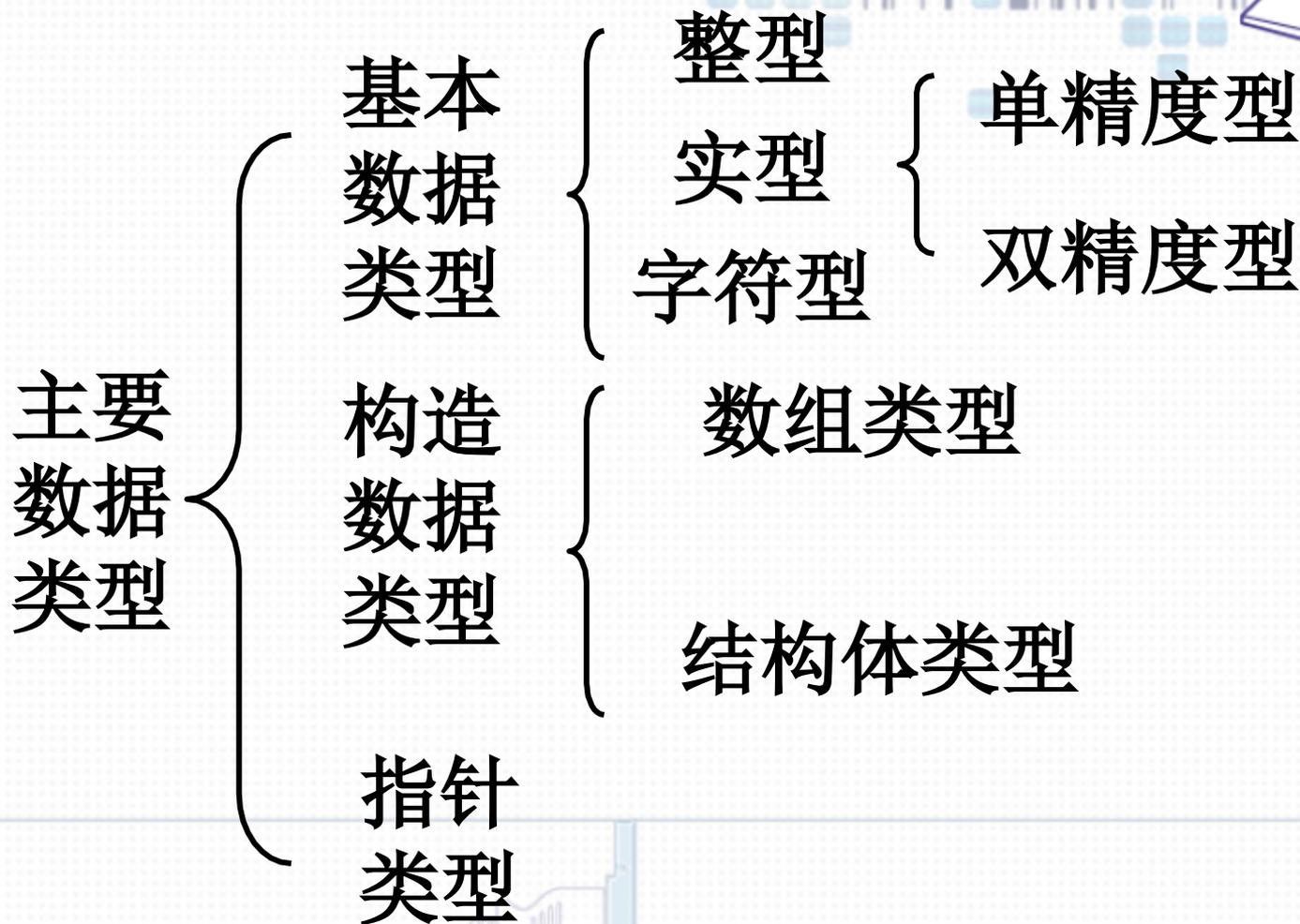
数据结构 + 算法 + 程序设计方法 + 语言工具



# 任务2 基本数据类型



## 一、C语言基本数据类型



# 任务2 基本数据类型

## 一、C语言基本数据类型

### 1、整型：

整型数据	说明符	字节数	数值范围
基本整型	int	4	$-2^{31} \sim (2^{31}-1)$ 即-2147483648 ~2147483647
短整型	short [int]	2	$-2^{15} \sim (2^{15}-1)$ 即-32768~ 32767
长整型	long [int]	4	$-2^{31} \sim (2^{31}-1)$ 即-2147483648 ~2147483647
无符号整型	unsigned [int]	4	$0 \sim (2^{32}-1)$ 即4294967295
无符号短整型	unsigned short [int]	2	$0 \sim (2^{16}-1)$ 即65535
无符号长整型	unsigned long [int]	4	$0 \sim 2^{32}-1$ 即4294967295

# 任务2 基本数据类型

## 一、C语言基本数据类型

### 1、整型：

#### 【例2-1】数据溢出的问题举例

```
#include <stdio.h>
int main()
{
    short int  a,b; /* 定义为2个字节的整型变量 */
    a=32767;
    b=a+1;
    printf(“ %d\n”,b);
    return 0;
}
```

# 任务2 基本数据类型



## 知识回顾

### 1) 、数据在计算机中是以二进制形式存储的

- 计算机内部的信息都是用二进制来表示的。
- 二进制数的特点是“逢二进一”。每一位的值只有0和1两种可能。

例如：十进制数10，用二进制表示是1010。它的含义是：

$$1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 0 \times 2^0$$

每一个二进制位代表不同的幂，最右边一位代表2的0次方，最右边第二位代表2的1次方，以此类推。显然一个很大的整数可能需要几十个“二进制位”来代表

# 任务2 基本数据类型



## 2) 、几个名词：位、字节和地址

**位**：又称“比特” (bit)。每一个二极管元件称为一个“二进制位”，是存储信息的最小单位。它的值是“1”或“0”。

**字节**：又称“拜特” (byte)。一般将8个“二进制位”组织成一组，称为“字节”。

**地址**：计算机的存储器包含许多存储单元，操作系统把所有存储单元以字节为单位编号。

# 任务2 基本数据类型



## 整型数据在内存中的存放形式

在C语言中为了能把加减法都统一成一种运算，对于整数，是以二进制补码 (complement) 来表示的。正数的补码是其本身。负数的补码是其绝对值按位取反一，再加1

以 -5 为例来进行说明吧。

第一步： -5 是一个负数，按照 (b) 规定来进行；

第二步： -5 的绝对值为 5，按照 16 位原码（例题中 short int 是 2 个字节）表示形式如下：

$|-5| = 5 = 0000\ 0000\ 0000\ 0101$

第三步：给  $|-5|$  的每位取反，结果如下所示：

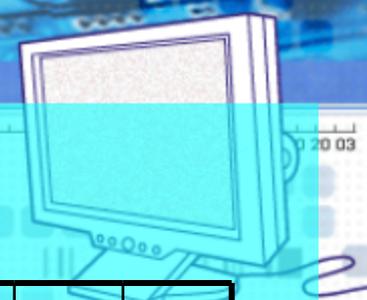
$|-5|$  取反 =  $1111\ 1111\ 1111\ 1010$

第四步：给  $|-5|$  取反的结果再加 1：

-5 的补码 =  $|-5|$  取反 + 1 =  $1111\ 1111\ 1111\ 1011$



# 任务2 基本数据类型



数据的溢出问题的解释:

32767换算成16位二进制是 0111 1111 1111 1111

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
														加	1
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

从前面的规定我们知道这个二进制数就是负数了。既然是负数就按负数的补码来还原：第一步：将1000 0000 0000 0000 减1

等于：

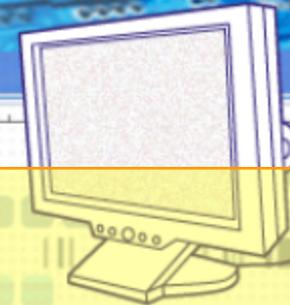
0111 1111 1111 1111

第二步按位取反

1000 0000 0000 0000

这个二进制数转换成十进制就是 $2^{15}=32768$ 。这个结果是某负数的绝对值。故程序在实际屏幕上显示 **-32768**。

# 任务2 基本数据类型



## 2、实型

实型也有常量和变量之分。

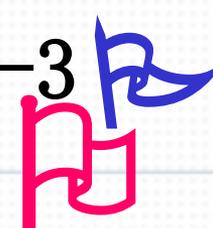
对于实型常量有二种形式：

- 1) 小数形式 例：3.1415 、 ~~0.~~123 0.0045
- 2) 指数形式 例：数学中的 $4.23 \times 10^5$ ，在C语言中表示为4.23E5或4.23e5

注意：字母e(或E)之前必须有数字，且e后面的指数必须为整数。

1e3、1.8e-3、-123e-6、-.1e-3

e3、2.1e3.5、.e3、e



## 实型变量的定义和分类

浮点型变量分为单精度（float型）、双精度（double型）和长双精度型（long double）三类形式。

类型	位数	数的范围	有效数字
float	32	$10^{-37} \sim 10^{38}$	6~7 位
double型	64	$10^{-307} \sim 10^{308}$	15~16位
long double	128	$10^{-4931} \sim 10^{4932}$	18~19位

**注意：实型数据输出时，一般用%f.**

# 任务2 基本数据类型

## 3、字符型

计算机诞生于美国，早期的计算机使用者大多使用英文，上世纪60年代，美国制定了一套英文字符与二进制位的对应关系，称为美国标准信息交换码（American Standard Code for Information Interchange），简称ASCII码，沿用至今。

ASCII码规定了128个英文字符与二进制的对应关系，占用一个字节（实际上只占用了个字节的后面7位，最前面1位统一规定为0）。例如，字母 a 的的ASCII码为 0110 0001（十进制数是97），那么你暂时可以理解为字母 a 存储到内存之前会被转换为 0110 0001，读取时遇到 0110 0001 也会转换为 a。

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/048061007066006124>