

第1章 程序设计和C语言

主要内容

1.1 计算机与程序、程序设计语言

1.2 C语言的发展及其特点

1.3 简单的C语言程序

1.4 C语言程序的结构

1.5 运行C程序的步骤与方法

1.6 程序设计的任务

1.1 计算机与程序、程序设计语言

➤ 计算机五大部件：

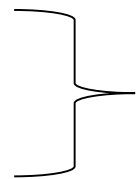
◆ 输入设备：键盘、鼠标

◆ 输出设备：显示器

◆ 存储器：硬盘、内存

◆ 运算器

◆ 控制器



CPU（中央处理单元）



1.1 计算机与程序、程序设计语言

- **程序**：一组计算机能识别和执行的**指令**
- 只要让计算机执行这个程序，计算机就会**自动地、有条不紊地**进行工作
- 计算机的一切操作都是由**程序**控制的，离开程序，计算机将一事无成
- 程序装载在内存中，**CPU**从内存中读入一条一条指令，执行控制、运算操作。



什么是程序设计语言

- **计算机语言**：人和计算机交流信息的、计算机和人都能识别的语言



什么是程序设计语言











低级语言

➤ 计算机语言发展阶段：

- ◆ 机器语言（由**0**和**1**组成的指令）
- ◆ 符号语言（用英文字母和数字表示指令）
- ◆ 高级语言（接近于人的自然语言和数学语言）
 - 面向**过程**的语言
（非结构化的语言、结构化语言）
 - 面向**对象**的语言 下学期的**C++**



为什么选C语言

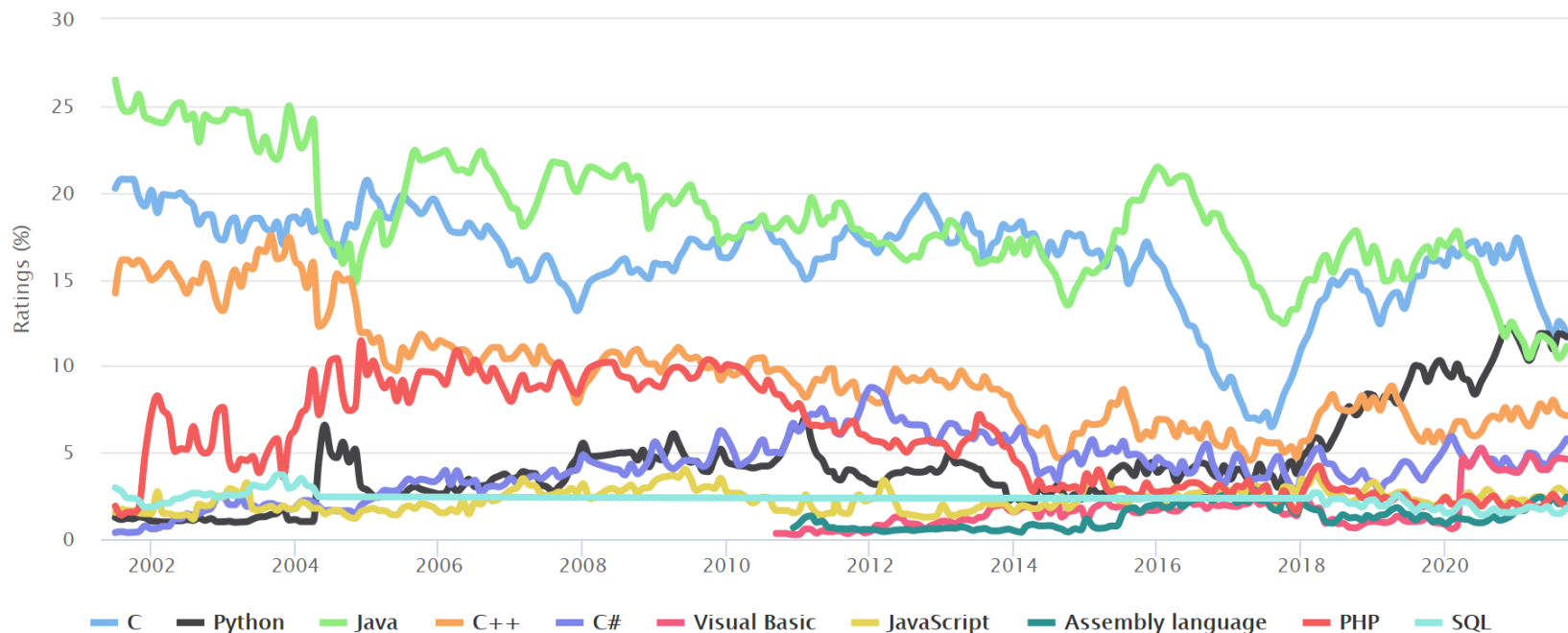
Sep 2021	Sep 2020	Change	Programming Language	Ratings	Change
1	1		 C	11.83%	-4.12%
2	3	▲	 Python	11.67%	+1.20%
3	2	▼	 Java	11.12%	-2.37%
4	4		 C++	7.13%	+0.01%
5	5		 C#	5.78%	+1.20%
6	6		 Visual Basic	4.62%	+0.50%
7	7		 JavaScript	2.55%	+0.01%
8	14	▲	 Assembly language	2.42%	+1.12%
9	8	▼	 PHP	1.85%	-0.64%
10	10		 SQL	1.80%	+0.04%

简洁、高效、基础、主流

为什么选C语言

TIOBE Programming Community Index

Source: www.tiobe.com

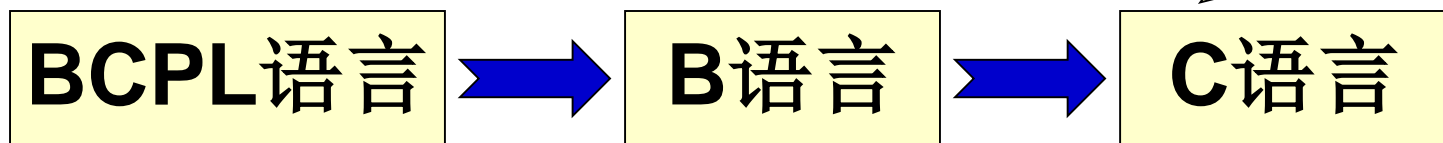


简洁、高效、基础、主流

1.2 C语言的发展及其特点

➤ **C语言**是国际上广泛流行的计算机高级语言。

➤ **C语言的发展**:



具有多种数据类型

精练、接近硬件，但过于简单，无数据类型



1.2 C语言的发展及其特点

- **1972-1973**年间，贝尔实验室的Dennis Ritchie发明了**C**语言。



1983,因发明Unix获得图灵奖

1999,因发明Unix和C语言获得
美国国家技术奖章

1.2 C语言的发展及其特点

- 最初的**C**语言只是为描述和实现**UNIX**操作系统提供一种工作语言而设计的。



1.2 C语言的发展及其特点

- **1983**年，美国国家标准协会(**ANSI**)成立了一个委员会，根据**C**语言问世以来各种版本对**C**语言的发展和扩充，制定了第一个**C**语言标准草案(**'83 ANSI C**)。



1.2 C语言的发展及其特点

- **1989年，ANSI公布了一个完整的C语言标准—ANSI X3.159-1989(常称ANSI C，或C89)。**



1.2 C语言的发展及其特点

- **1990年**，国际标准化组织
ISO(International Standard Organization) 接受**C89**作为国际标准
ISO/IEC 9899:1990，它和
ANSI的C89基本上是相同的。



1.2 C语言的发展及其特点

- **1995年，ISO对C90作了一些修订，1999年，ISO又对C语言标准进行修订，在基本保留原来的C语言特征的基础上，针对应用的需要，增加了一些功能，尤其是C++中的一些功能，命名为ISO/IEC 9899:1999。**



1.2 C语言的发展及其特点

- **2001、2004**年先后进行了两次技术修正（**TC1**和**TC2**）。

ISO/IEC 9899:1999(及其技术修正)
被称为 **C99**。

- **C99**是**C89**(及**1995**基准增补**1**)的扩充。



1.2 C语言的发展及其特点

- 本书的叙述以**C99**标准为依据（对**C99**新增加的功能作特别的说明）。
- 目前不同软件公司提供的各**C**语言编译系统多数并未完全实现**C99**建议的功能
- 本书中程序基本上都可以在目前所用的编译系统(如**VC++ 6.0**, **Turbo C++ 3.0**, **GCC**)上编译和运行。



1.2 C语言的发展及其特点

- **C语言**是一种用途广泛、功能强大、使用灵活的过程性(**procedural**)编程语言，既可用于编写应用软件，又能用于编写系统软件。因此**C语言**问世以后得到迅速推广。



1.2 C语言的发展及其特点

➤ C语言主要特点:

◆语言简洁、紧凑，使用方便、灵活。

●只有**37**个关键字、**9**种控制语句

●程序书写形式自由，源程序短



1.2 C语言的发展及其特点

➤ C语言主要特点:

◆ 运算符丰富。

- 有**34**种运算符
- 把括号、赋值、强制类型转换等都作为运算符处理
- 表达式类型多样化



1.2 C语言的发展及其特点

➤ C语言主要特点:

◆数据类型丰富。

- 包括:整型、浮点型、字符型、数组类型、指针类型、结构体类型、共用体类型
- C99**又扩充了复数浮点类型、超长整型(**long long**)、布尔类型(**bool**)
- 指针类型数据, 能用来实现各种复杂的数据结构(如链表、树、栈等)的运算。



1.2 C语言的发展及其特点

➤ C语言主要特点:

◆ 具有结构化的控制语句

- 如**if...else**语句、**while**语句、**do...while**语句、**switch**语句、**for**语句
- 用函数作为程序的模块单位，便于实现程序的模块化
- C语言是完全模块化和结构化的语言



1.2 C语言的发展及其特点

➤ C语言主要特点:

◆语法限制不太严格，程序设计自由度大。

- 对数组下标越界不做检查

- 对变量的类型使用比较灵活，例如，整型量与字符型数据可以通用

- C语言允许程序编写者有较大的自由度，因此放宽了语法检查



C语言混乱代码大赛

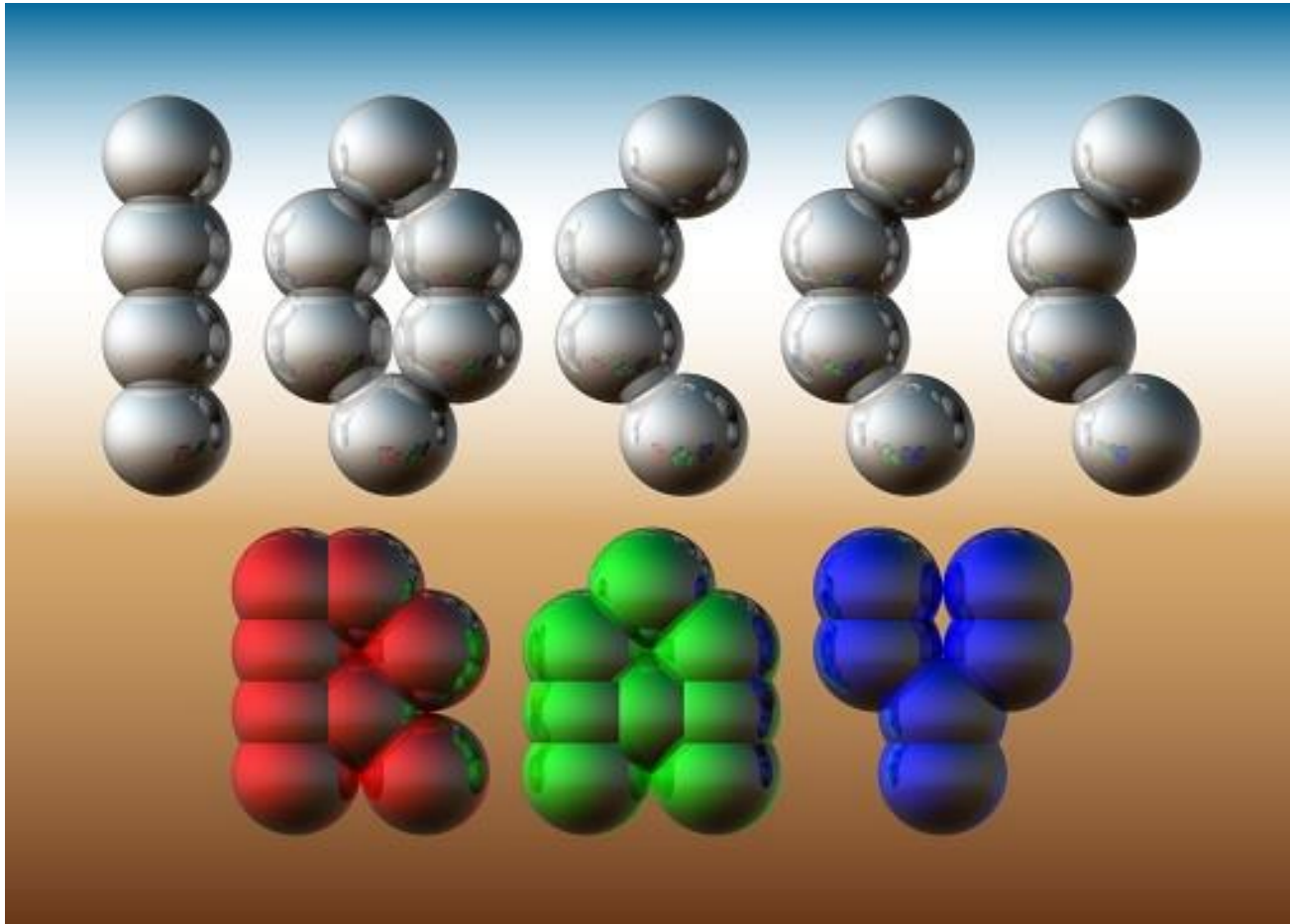
```

X=400; Y=300; A=5;

J=0;K=-10;L=-7;M=1296;N=36;O=255;P=9;_1<<15;E:S;C;D;F(b){E="1""111886:6:??AAF"
"FGHMM005555779988>>>BBBGGIIKK"[b]-64;C="C8@=:C8@=:C8@=:C8@=:C5""31/513/5131/"
"31/531/53"[b]-64;S=b<22?9:0;D=2;I(x,Y,X){Y?(X^=Y,X*>X?(X^=Y):0,I(x,Y/2,X
)}:(E=X);H(x){I(x,_,0);p;q(c,x,y,z,k,l,m,a,b){F(c
):x-=E*M;y-=S*M;z-=C*M;b=x*x/M+y*y/M+z
*z/M-D*D*M;a=-x*k/M-y*l/M-z*m/M;p=((b=a*a/M-
b)>0?(I(b*M,_,0),b=E,a+(a>b?-b:b)):-1.0);Z;W;o
(c,x,y,z,k,l,m,a){Z!=c?-1:Z;c<44?(q(c,x,y,z,k,
l,m,0,0),(p>0&&c!=a&&(p<W||Z<0))?(W=
p,Z=c):0,o(c+l,x,y,z,k,l,m,a)):0;Q;T:
U;u;v;w;n(e,f,g,h,i,j,d,a,b,V){o(0,e,f,g,h,i,j,a);d>0
&&Z>0?(e+=h*W/M,f+=i*W/M,g+=j*W/M,F(Z),u=e-E*M,v=f-S*M,w=g-C*M,b=(-2*u-2*v+w)
/3,H(u*u+v*v+w*w),b/=D,b*=b,b*=200,b/=(M*M),V=Z,E!=0?(u=-u*M/E,v=-v*M/E,w=-w*M/
E):0,E=(h*u+i*v+j*w)/M,h=-u*E/(M/2),i=-v*E/(M/2),j=-w*E/(M/2),n(e,f,g,h,i,j,d-1
,Z,0,0),Q/=2,T/=2,U/=2,V=V<22?7:(V<30?1:(V<38?2:(V<44?4:(V==44?6:3))))
,Q+=V&1?b:0,T+=V&2?b:0,U+=V&4?b:0:(d==P?(g+=2
,j)=g>0?g/8:g/20):0,j>0?(U=j*j/M,Q=255-250*U/M,T=255
-150*U/M,U=255-100*U/M):(U=j*j/M,U<M/5?(Q=255-210*U
/M,T=255-435*U/M,U=255-720*U/M):(U=-M/5,Q=213-110*U
/M,T=168-113*U/M,U=111-85*U/M),d!=P?(Q/=2,T/=2
,U/=2):0;Q=Q<0?0:Q>0?0:Q;T=T<0?0:T>0?0:T;U=U<0?0:
U>0?0:U;R;G;B;t(x,y,a,b){n(M*J+M*40*(A*x+a)/X/A-M*20,M*K,M
*L-M*30*(A*y+b)/Y/A+M*15,0,M,0,P,-1,0,0);R+=Q;G+=T;B+=U;
++a<A?t(x,y,a,b):(++b<A?t(x,y,0,b):0);r(x,y){R=G+B=0;t(x,y,0,0);x<X?(printf("%c%c%c",R/A,A,G
/A/A,B/A/A),r(x+1,y)):0;}s(y){r(0,--y?s(y),y:y);}main(){printf("P6\n%i%i\n255"
"\n",X,Y);s(Y);}

```


IOCCC_RAY



1.2 C语言的发展及其特点

➤ C语言主要特点:

- ◆ 允许直接访问物理地址，能进行位操作，可以直接对硬件进行操作

- C语言具有高级语言的功能和低级语言的许多功能，可用来编写系统软件

- 这种双重性，使它既是成功的系统描述语言，又是通用的程序设计语言



1.2 C语言的发展及其特点

➤ C语言主要特点:

◆用C语言编写的程序可移植性好。

- C的编译系统简洁，很容易移植到新系统
- 在新系统上运行时，可直接编译“标准链接库”中的大部分功能，不需要修改源代码
- 几乎所有计算机系统都可以使用C语言



1.2 C语言的发展及其特点

➤ **C语言**主要特点:

◆ 生成目标代码质量高，程序执行效率高。



小测试

➤ **C语言**是一种（）

A 符号语言 **B** 机器语言 **C** 低级语言 **D** 高级语言

➤ 最初的**C语言**是为描述和实现哪种操作系统提供一种工作？

A Unix **B Windows** **C Linux** **D Word**

小测试

➤ **C**语言之父是（ ）

A Barack Obama **B** 谭浩强

C Dennis Ritchie **D** Bill Gates

1.3 简单的C语言程序举例

例**1.1** 要求在屏幕上输出以下一行信息。

Hello world.

➤ 解题思路：

在主函数中用**printf**函数原样输出以上文字。



1.3 简单的C语言程序举例

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

```
int main( )
```

C程序必须有一个 main 函数

```
{
```

函数的名字，表示主函数

```
printf ("Hello world.\n");
```

```
return 0;
```

```
}
```



1.3 简单的C语言程序举例

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

```
int main( )
```

```
{
```

主函数类型

```
    printf ("Hello world.\n");
```

```
    return 0;
```

```
}
```



1.3 简单的C语言程序举例

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

```
int main( )
```

```
{
```

```
    printf ("Hello world.\n");
```

```
    return 0;
```

```
}
```

函数体



1.3 简单的C语言程序举例

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

```
int main( )
```

```
{
```

输出函数

```
printf ("Hello world.\n");
```

```
return 0;
```

```
}
```

输出语句



1.3 简单的C语言程序举例

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

```
int main( )
```

```
{
```

```
    printf ("Hello world.\n");
```

```
    return 0;
```

```
}
```

换行符



1.3 简单的C语言程序举例

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

```
int main( )
```

```
{
```

```
    printf (" Hello world.\n");
```

```
    return 0;
```

```
}
```

当main函数执行结束前
将整数0作为函数值



1.3 最简单的C语言程序举例

#include <stdio.h>

int main()

{

printf ("Hello world.\n");

return 0;

}

用到函数库中的输入输出函数时

表示语句结束



1.3 最简单的C语言程序举例

C语言允许用两种注释方式：

➤ **//**：单行注释

◆可单独占一行

◆可出现在一行中其他内容的右侧

➤ **/*.....*/**：块式注释

◆可包含多行



例1.2 求两个整数之和。

➤ 解题思路：

- ◆ 设置3个变量

- ◆ **a**和**b**用来存放两个整数

- ◆ **sum**用来存放和数

- ◆ 用赋值运算符“=”把结果传送给**sum**




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

```
int main( )
```

```
{
```

```
    int a,b,sum;    定义整型变量a,b,sum
```

```
    a = 123; } 对变量a,b赋值
```

```
    b = 456;
```

```
    sum = a + b; 将a与b的和赋给sum
```

```
    printf("sum is %d\n",sum);
```

```
    return 0;
```

```
}
```

```
sum is 579
```



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

```
int main( )
```

```
{
```

```
    int a,b,sum;
```

```
    a = 123;
```

```
    b = 456;
```

```
    sum = a + b;
```

```
    printf("sum is %d\n",sum);
```

```
    return 0; 希望输出的字符
```

```
}
```

用sum的值替代

sum is	579
--------	-----



例**1.3**求两个整数中的较大者。

➤解题思路:

- ◆用一个函数实现求两个整数中的较大者
- ◆在主函数中调用此函数并输出结果



C程序设计

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

```
int main( )
```

```
{
```

```
    int max(int x,int y);
```

```
    int a,b,c;
```

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

```
    c = max(a,b);
```

```
    printf("max=%d\n",c);
```

```
    return 0;
```

```
}
```

主函数

max函数

```
int max(int x,int y)
```

```
{
```

```
    int z;
```

```
    if (x > y) z = x;
```

```
    else z = y;
```

```
    return(z);
```

```
}
```



C程序设计

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

```
int main( )
```

```
{
```

```
    int max(int x,int y);
```

```
    int a,b,c;
```

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

```
    c = max(a,b);
```

```
    printf("max=%d\n",c);
```

```
    return 0;
```

```
}
```

将x和y中较大者
值返回给主函数

```
int max(int x,int y)
```

```
{
```

```
    int z;
```

```
    if (x > y) z = x;
```

```
    else z = y;
```

```
    return(z);
```

```
}
```



C程序设计

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

```
int main( )
```

```
{
```

```
    int max(int x,int y);
```

```
    int a,b,c;
```

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

```
    c = max(a,b);
```

```
    printf("max=%d\n",c);
```

```
    return 0;
```

```
}
```

```
int max(int x,int y)
```

```
{
```

```
    int z;
```

```
    if (x > y) z = x;
```

```
    else z = y;
```

```
    return(z);
```

```
}
```



C程序设计

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

```
int main( )
```

```
{ 因max函数的定义在main函数之后，需声明
```

```
int max(int x,int y);
```

```
int a,b,c;
```

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

```
c = max(a,b);
```

```
printf("max=%d\n",c);
```

```
return 0;
```

```
}
```

```
int max(int x,int y)
```

```
{
```

```
int z;
```

```
if (x > y) z = x;
```

```
else z = y;
```

```
return(z);
```

```
}
```



C程序设计

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

```
int main( )
```

```
{
```

```
    in  输入函数 (int x,int y);
```

```
    int a,b,c;
```

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

```
    c = max(a,b);
```

```
    printf("max=%d\n",c);
```

```
    return 0;
```

```
}
```

```
int max(int x,int y)
```

```
{
```

```
    int z;
```

```
    if (x > y) z = x;
```

```
    else z = y;
```

```
    return(z);
```

```
}
```



C程序设计

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

```
int main( )
```

```
{
```

```
int max(
```

输入语句

```
int a,b,c;
```

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

```
c = max(a,b);
```

```
printf("max=%d\n",c);
```

```
return 0;
```

```
}
```

8,5

```
int max(int x,int y)
```

```
{
```

```
int z;
```

```
if (x > y) z = x;
```

```
else z = y;
```

```
return(z);
```

```
}
```



C程序设计

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

```
int main( )
```

```
{
```

```
int x, y);
```

```
int a,b,c;
```

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

```
c = max(a,b);
```

```
printf("max= ");
```

```
return 0;
```

```
}
```

输入的数据
放到a,b中

输入格式

a的地址

8,5

```
int max(int x,int y)
```

```
{
```

```
int z;
```

```
if (x > y) z = x;
```

```
else z = y;
```

```
return(z);
```

```
}
```



C程序设计

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

```
int main( )
```

```
{
```

```
    int max(int x,int y);
```

调用max函数

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

```
    c = max(a,b);
```

```
    printf("max=%d\n",c);
```

```
    return 0;
```

```
}
```

8,5

```
int max(int x,int y)
```

```
{
```

```
    int z;
```

```
    if (x > y) z = x;
```

```
    else z = y;
```

```
    return(z);
```

```
}
```



C程序设计

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

```
int main( )
```

```
{
```

```
    int max(int x,int y);
```

```
    int a,b,c;
```

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

```
    c = max(a,b);
```

```
    printf("max=%d\n",c);
```

```
    return 0;
```

```
}
```

8,5

8
↓

5
↓

```
int max(int x,int y)
```

```
{
```

```
    int z;
```

```
    if (x > y) z = x;
```

```
    else z = y;
```

```
    return(z);
```

```
}
```

8



C程序设计

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

```
int main( )
```

```
{
```

```
    int max(int x,int y);
```

```
    int a,b,c;
```

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

```
    c = max(a,b);
```

```
    printf("max=%d\n",c);
```

```
    return 0;
```

```
}
```

```
8,5  
max=8
```

8
↓

5
↓

```
int max(int x,int y)
```

```
{
```

```
    int z;
```

```
    if (x > y) z = x;
```

```
    else z = y;
```

```
    return(z);
```

```
}
```

8



C程序设计

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

```
int main( )
```

```
{
```

```
int max(int x, int y);
```

```
int a,
```

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

```
c = max(a,b);
```

```
printf("max=%d\n",c);
```

```
return 0;
```

```
}
```

实际参数

```
8,5  
max=8
```

形式参数

```
int max(int x,int y)
```

```
{
```

```
int z;
```

```
if (x > y) z = x;
```

```
else z = y;
```

```
return(z);
```

```
}
```



1.4 C语言程序的结构

C语言程序的结构特点：

1. 一个程序由一个或多个源程序文件组成

◆ 小程序往往只包括一个源程序文件

◆ **例1.1**， **例1.2** 只有一个函数

◆ **例1.3** 有两个函数

只包括一个源程序文件



1.4 C语言程序的结构

C语言程序的结构特点：

➤ 一个源程序文件中可以包括三个部分：

- ◆ 预处理指令 `#include <stdio.h>`等
- ◆ 全局声明 在函数之外进行的数据声明
- ◆ 函数定义 每个函数用来实现一定的功能



1.4 C语言程序的结构

C语言程序的结构特点：

2. 函数是C程序的主要组成部分

- ◆ 一个C程序是由一个或多个函数组成的
- ◆ 必须包含一个**main**函数（只能有一个）
- ◆ 每个函数都用来实现一个或几个特定功能
- ◆ 被调用的函数可以是库函数，也可以是自己编制设计的函数



1.4 C语言程序的结构

C语言程序的结构特点：

3. 一个函数包括两个部分：

◆ 函数首部

函数的第1行

int

max

(int x, int y)

函数类型

函数名

参数类型

参数名



1.4 C语言程序的结构

C语言程序的结构特点：

3. 一个函数包括两个部分：

◆ 函数首部

int max (int x, int y)

若函数无参，在括弧中写**void**或空括弧

int main(void) 或 int main()



1.4 C语言程序的结构

C语言程序的结构特点：

3. 一个函数包括两个部分：

◆ 函数体

● 声明部分

可以没有声明部分

★ 定义在本函数中所用到的变量

★ 对本函数所调用函数进行声明

● 执行部分：由若干个语句组成，指定在函数中所进行的操作



1.4 C语言程序的结构

C语言程序的结构特点：

3. 一个函数包括两个部分：

◆ 函数体

可以是空函数

```
void dump ( )  
{  
}
```



1.4 C语言程序的结构

C语言程序的结构特点：

4. 程序总是从**main**函数开始执行

5. C程序对计算机的操作由C语句完成

◆C程序书写格式是比较自由的

- 一行内可以写几个语句

- 一个语句可以分写在多行上

◆为清晰起见，习惯上每行只写一个语句



1.4 C语言程序的结构

C语言程序的结构特点：

- 4. 程序总是从main函数开始执行**
- 5. C程序对计算机的操作由C语句完成**
- 6. 数据声明和语句最后必须有分号**
- 7. C语言本身不提供输入输出语句**
- 8. 程序应当包含注释，增加可读性**



1.5 运行C程序的步骤与方法

1. 上机输入和编辑源程序（**.c**文件）
2. 对源程序进行编译（**.obj**文件）
3. 进行连接处理（**.exe**文件）
4. 运行可执行程序，得到运行结果



1.6 程序设计的任务

1. 问题分析

- 对于接手的任务要进行认真的分析
- 研究所给定的条件
- 分析最后应达到的目标
- 找出解决问题的规律
- 选择解题的方法



1.6 程序设计的任务

1.问题分析

2.设计算法

➤ 设计出解题的方法和具体步骤



1.6 程序设计的任务

1.问题分析

2.设计算法

3.编写程序

4.对源程序进行编辑、编译和连接

5.运行程序，分析结果

◆结果错了，程序肯定错

◆结果对了，程序未必对



1.6 程序设计的任务

1. 问题分析
2. 设计算法
3. 编写程序
4. 对源程序进行编辑、编译和连接
5. 运行程序，分析结果
6. 编写程序文档

