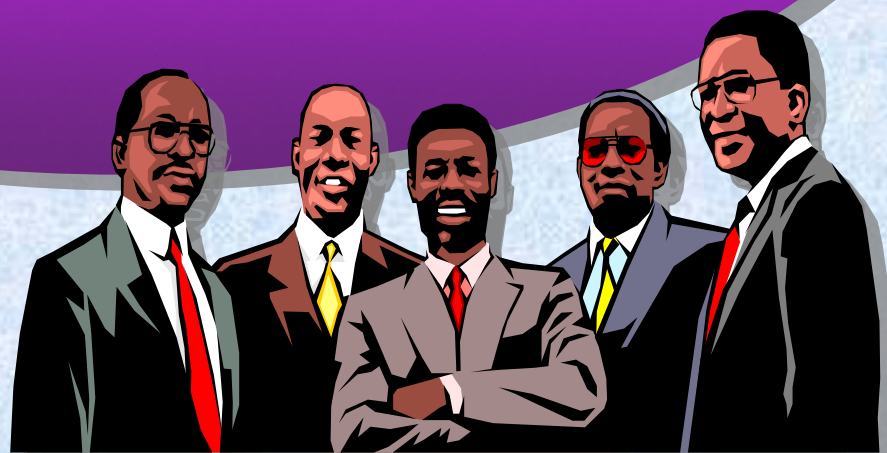
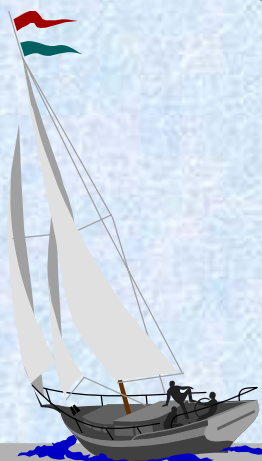


第2讲 I/O输入输出

深圳大学计算机系



2011-10-10

→第2讲

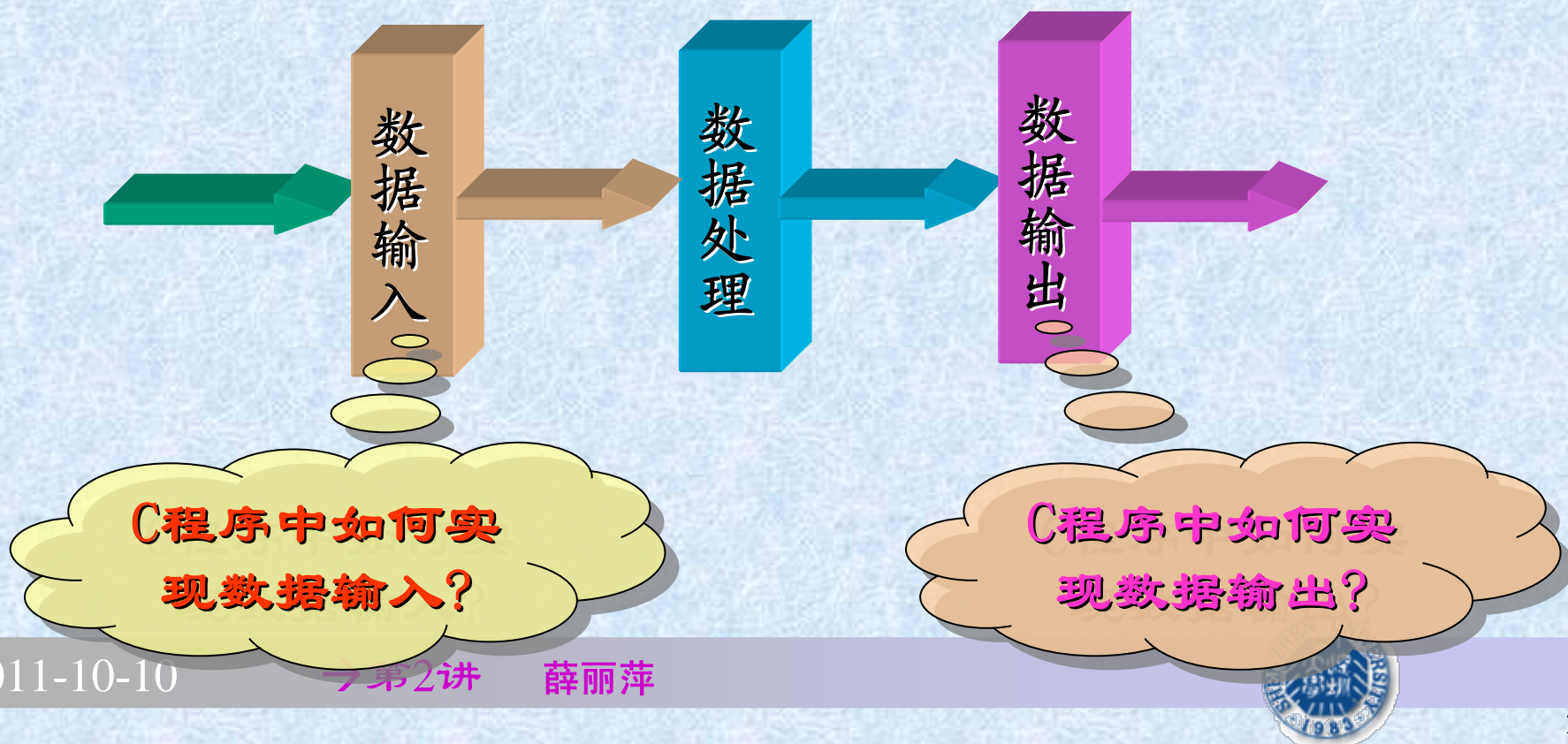
薛丽萍



◆ 学习的意义

程序对数据的处理流程：

程序的主要功能就是对数据的处理，其整个流程主要包括数据的输入、数据的处理、数据的输出。



C程序 ---- 输出一条信息

/*最简单的程序示例 */

#include <iostream.h>

//预处理命令 **iostream**

void main()

{

cout<<“Hello World!”<<endl;

}

作用等同于“\n”，
不过“\n”是C语言
的换行符，在
C++语言中保留。

运行结果: Hello World!



转义字符及其含义：P18

`\n` ---- 回车换行

`\t` ---- 水平制表

`\b` ---- 退格

`\f` ---- 换页

`\?` ---- 打印问号

`\'` ---- 打印单引号

`\v` ---- 垂直制表

`\r` ---- 回车

`\a` ---- 响铃报警

`\\` ---- 打印反斜线

`\“` ---- 打印双引号



练习

1.编写程序，输出姓名和学号。



练习

1.编写程序，输出姓名和学号。 `/*exc21.cpp*/`

- `#include <iostream.h>`
- `void main()`
- `{`
- `cout<<"学号:20110001011\n";`
- `cout<<"姓名:王小二 \a"<<endl;`
- `}`



int,cin,cout,printf



基本输入

可以利用 `cin` 后跟 “>>” (两个“>”号) 来给 “>>” 右边的数据输入值。



int,cin,cout,printf

例： 用一个程序实现求一个表达式 $4a-3b+2$ 的值

`/*exc21.cpp*/`

```
#include <iostream.h>
```

```
void main()
```

```
{
```

```
    int a,b,result;
```

```
    cout<<"please input two integers:"; //要有提示
```

```
    cin>>a>>b;
```

```
    result=4*a-3*b+2;
```

```
    cout<<"the result is:"<<result<<endl;
```

```
}
```



变量和标识符

1. 标识符

- **定义：**用来标识变量、常量、函数等的字符序列
- **组成：**
 - 只能由**字母**、**数字**、**下划线**组成，且第一个字母必须是字母或下划线
 - C语言的关键字不能
 - 大小写敏感

• 长个

char是关

>字符非法

.字

数字3不可作首字符

\$不可作首字符

判断标识符符号合法性
m Sn M.D.John day Date 3days
student_name #33 lotus_1_2_3
char a>b _above \$123



变量和标识符

2. 变量

- 定义：程序运行时其值可以被改变的量
- 变量的两要素：变量名、变量值

short int a=1, b=-3, c;

地址

例2:

```
float a,b,c;
```

```
c = a % b; //Illegal use
```

```
void main ( )
```

```
{ { int a, b = 2;
```

```
  a = 1;
```

```
  float data;
```

```
  data = (a + b) * 1.2;
```

```
  printf("data=%f\n",data);
```

```
} }
```

变量命名

P15

- 命名规则：只能包含字符、数字、下划线，第一个字符必须是字符和下划线。
- 大小写敏感。如temp,Temp是两个不同变量。
- 不能是C的关键字（也叫保留字）。如void ,main等。



- **变量名的骆驼表示法:**

以小写字母开头,以后的单词都以大写字母开头.

如: **myBook** **myCar** **sizeOfChaaar**



- 变量名的匈牙利命名法:

变量名: **属性 + 类型 + 对象**

属性: 后面讲解

类型: 后面讲解

对象: 通过变量名可知道它所表示的数据含义, 常选择对应的英文单词做变量名。即见名知义。

如: iHeight, iWidth, fpArea



练习

1. 编程，输入长方形边长 a, b ，输出面积和周长。
2. 编程，输入长方体的三条边长 a, b, c ，输出体积和表面积。/*上机练习*/



练习

1. 编程，输入长方形边长a,b，输出面积 和周长。/*exc23.cpp*/

- **#include <iostream.h>**
- **void main()**
- **{**
- **int a,b,area,roundLength;**
- **//cout<<"please input length and width:";**
- **cin>>a>>b;**
- **area=a*b;**
- **roundLength=2*(a+b);**
- **cout<<"the area is:"<<area<<endl;**
- **cout<<"the roundLength is:"<<roundLength<<endl;**
- **}**



格式化输出printf

#include <stdio.h>

➤ 一般格式

```
printf ("格式控制字符串", 表达式1, 表达式2, ..., 表达式n);
```

➤ 功能

按照“格式控制字符串”的要求，将表达式1，表达式2，...，表达式n的值显示在计算机屏幕上。

➤ 说明

格式控制字符串用于指定输出格式。它包含两类字符：

常规字符：包括可显示字符和格式控制符的字符。

格式控制符：以%开头的字符，说明输出数据的类型、形式、长度等。其格式为：

常规字符

格式控制符
(修饰符)

格式控制符
(格式转换符)

常规字符
(转义符)

例：
long int x = 0x8AB6;
printf (“The Value of x is %ld\n”, x);

➤ printf使用时的注意事项

(1) 格式控制字符串可以不包含任何格式控制符。

```
printf ("how are you?\n");           //只有一个字符串参数,  
                                     输出为: how are you?  
printf ("how old are you?\n", 20);  //带有两个参数, 20没有意义,  
                                     输出为: how old are you?
```

到右的格式控制符对应从左到右的表达式。

(3) 已知: `int a = 2;` 达

```
printf ("5 + 3 = %d, 5 - 3 = %d, 5 * 3 = %d", 5 + 3, 5 - 3);  
输出结果将是: 5 + 3 = 8, 5 - 3 = 2, 5 * 3 = -28710
```

```
char ch = 'A';  
printf ("ch = %c", ch);  //输出结果: ch = A  
                        (以字符形式输出)  
printf ("ch = %d", ch);  //输出结果: ch = 65  
                        (以'A'字符的ASCII码形式输出)
```


➤ printf函数中的格式转换字符及其含义

格式转换字符	含 义	对应的表达式数据类型
%d 或 %i	以十进制形式输出一个整型数据。例如： int a = 20; printf ("%d", a); //输出20	有符号整型
%x, %X	以十六进制形式输出一个无符号整型数据。例如： int a = 164; printf ("%x", a); //输出a4 printf ("%X", a); //输出A4	无符号整型
%o (字母o)	以八进制形式输出一个无符号整型数据。例如： int a = 164; printf ("%o", a); //输出244	无符号整型
%u	以十进制形式输出一个无符号整型数据。例如： int a = -1; printf ("%u", a); //BC下输出65535, VC下4294967295	无符号整型
%c	输出一个字符型数据。例如： char ch = 'A'; printf ("%c", ch); //输出A	字符型



➤ printf函数中的格式转换字符及其含义

格式转换字符	含 义	对应的表达式数据类型
%s	输出一个字符串。例如： <code>printf("my name is %s", "wangjinhua");</code> //输出my name is wangjinhua	字符串
%f	以十进制小数形式输出一个浮点型数据。例如： <code>float f = -12.3;</code> <code>printf("%f", f);</code> //输出-12.300000	浮点型
%e, %E	以指数形式输出一个浮点型数据。例如： <code>float f = 1234.8998;</code> <code>printf("%e", f);</code> //输出1.234900e+003 <code>printf("%E", f);</code> //输出1.234900E+003	浮点型
%g, %G	按照%f或%e中输出宽度比较短的一种格式输出。	浮点型
%p	以主机的格式显示指针，即变量的地址。例如： <code>int a = 2;</code> <code>printf("%p", &a);</code> //BC下输出FFD8， VC下 0012FF7C	指针类型



【例】有符号整数的格式化输出

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

```
void main ( )
```

```
{  
    int a = 123;  
    long L = 65537;  
    printf (" 12345678901234567890\n");  
    printf ("a=%d-----(a=% %d)\n", a);  
    printf ("a=%6d-----(a=% %6d)\n", a);  
    printf ("a=%+6d-----(a=% %+6d)\n", a);  
    printf ("a=%-6d-----(a=% %-6d)\n", a);  
    printf ("a=%-06d-----(a=% %-06d)\n", a);  
    printf ("a=%+06d-----(a=% %+06d)\n", a);  
    printf ("a=%+6.6d-----(a=% %+6.6d)\n", a);  
    printf ("a=%6.6d-----(a=% %6.6d)\n", a);  
    printf ("a=%-6.5d-----(a=% %-6.5d)\n", a);  
    printf ("a=%6.4d-----(a=% %6.4d)\n", a);  
    printf ("L=%ld-----(L=% %ld)\n", L);  
    printf ("L=%hd------(L=% %hd)\n", L);  
}
```

L是一长整型数65537，
其值为十六进制
0X0001**0001**，所以要将
其转换成短整型，即取
低16位**0x0001**，将其输
出，故输出为1。

运行结果：

```
12345678901234567890  
a=123----- (a=%d)  
a=   123----- (a=%6d)  
a=  +123----- (a=%+6d)  
a=123      ----- (a=%-6d)  
a=123      ----- (a=%-06d)  
a=+00123----- (a=%+06d)  
a=+000123---- (a=%+6.6d)  
a=000123----- (a=%6.6d)  
a=00123  ----- (a=%-6.5d)  
a=  0123----- (a=%6.4d)  
L=65537----- (L=%ld)  
L=1----- (L=%hd)
```

Why?

格式化输入scanf

➤ 一般格式

scanf (“格式控制字符串”, 变量1的地址, 变量2的地址, ..., 变量n的地址);

➤ 功能

第一个参数格式控制字符串的控制下, 接受用户的键盘输入, 并将输入的数据依此存放在变量1、变量2、.....、变量n中。

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

➤ 格式控制符与后续参数中的变量地址的对应关系

已知: int a, b;

函数调用: scanf ("%d%d", &a, &b);

假设输入: 10□20✓ 变量的值: a的值是10, b的值是20

➤ 使用scanf函数注意事项

(1) 如果相邻两个格式控制符之间，不指定数据分隔符（如逗号、冒号等），则相应的两个输入数据之间，至少用一个空格分隔，或者用Tab键分隔，或者输入一个数据后，

例如：

```
scanf("%d%d", &num1, &num2);
```

假设给num1输入12，给num2输入36，则正确的输入操作为：

12□36✓

或者

12✓

36✓

使用“✓”符号表示按回车键操作，在输入数据操作中的作用是，通知系统输入操作结束。

例如：

```
scanf("%d:%d:%d", &h, &m, &s);
```

假设给h输入12，给m输入30，给s输入10，正确的输入操作为：

12:30:10✓

【例】数

输入一
F: 女) 及
总分和平均

```
#include  
void main  
{  
    unsigned  
    unsigned
```

```
printf ("input the student's Scores(chinese, math, english): ");  
scanf ("%f,%f,%f", &chinese, &math, &english);  
total = chinese + math + english; //计算总分  
average = total / 3; //计算平均分  
printf ("\n===NO=====birthday==sex==chinese==math==  
        english==total==average\n");  
printf ("%08ld  %4d-%02d-%02d  %c  %-5.1f %-5.1f  
        %-5.1f %-5.1f %-5.1f\n", no, year, month, day, sex,  
        chinese, math, english, total, average);
```

假设输入 :

```
input the student's NO: 20040101✓  
input the student's Birthday(yyyy-mm-dd): 1987-9-8✓  
input the student's Sex(M/F): M✓  
input the student's Scores(chinese, math, english): 90, 80, 90✓
```

输出结果 :

```
===NO=====birthday==sex==chinese==math==english==total==average  
20040101  1987-09-08  M      90.0    80.0    90.0    260.0    86.7
```


C语言编程习惯

- (1) 一行只放一条语句。
- (2) 养成随时给程序加注释的习惯。
- (3) 程序的书写要有层次感，该缩进的一定缩进。

- (4) 编写函数时，变量定义部分和函数的执行部分之间增

加
加
void main()

{

int i, sum;

- (5) sum=0;

for (i=1; i<=100; i++)

if (i%2 ==0) sum+=i;

printf("sum=%d", sum);

- (6) }

序
误
易

没有层次感

void main ()

{

int i, sum;

层次清晰

sum = 0;

for (i = 1; i <= 100; i++)

if (i % 2 == 0)

sum += i;

printf ("sum = %d", sum);

}



上机问题

- **error LNK2001: unresolved external symbol
_main**

主函数拼写错误;



- 关于输出printf:

printf(“格式控制字符串”, 表达式1, 表达式2, ..., 表达式n)

格式控制串中的格式控制符(%d,%f等) 必须与表达式一一对应 (个数和类型)。

- error C2065: 'v' : undeclared identifier
变量v必未定义。变量须先定义后使用
- 注意变量的类型与定义是否一致
- error LNK2005: _main already defined in test.obj

查看工程中的源程序, 会发现两个主函数。





希望大家能学出好成绩,我们一起努力!

谢谢大家!