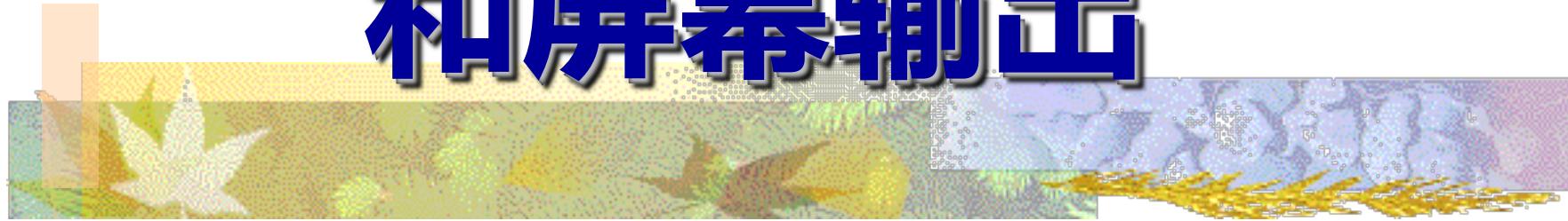




規格严格 功夫到家



第4章 键盘输入 和屏幕输出



哈尔滨工业大学

计算机科学与技术学院

苏小红

sxh@hit.edu.cn

本章学习内容

- ☞ 字符常量与转义字符
- ☞ 字符输出函数**putchar()**
- ☞ 字符输入函数**getchar()**
- ☞ 数据的格式化输出函数**printf()**
- ☞ 数据的格式化输入函数**scanf()**

字符串常量

■ 字符常量是用单引号括起来的一个字符

- 'a' 是字符常量，而 a 则是一个标识符
- '3' 表示一个字符常量，而 3 则表示一个整数

■ 转义字符 (Escape Character)

- 一些特殊字符（无法从键盘输入或者另有他用）用转义字符表示

字 符	含 义	字 符	含 义
'\n'	换行 (Newline)	'\a'	响铃报警提示音 (Alert or Bell)
'\r'	回车 (不换行) (Carriage Return)	'\"'	一个双引号 (Double Quotation Mark)
'\0'	空字符，通常用作字符串结束标志 (Null)	'\''	单引号 (Single Quotation Mark)
'\t'	水平制表 (Horizontal Tabulation)	'\\'	一个反斜线 (Backslash)
'\v'	垂直制表 (Vertical Tabulation)	'\?'	问号 (Question Mark)
'\b'	退格 (Backspace)	'\ddd'	1 到 3 位八进制 ASCII 码值所代表的字符
'\f'	走纸换页 (Form Feed)	'\xhh'	1 到 2 位十六进制 ASCII 码值所代表的字符

字符串常量

- 字符型变量的取值范围取决于计算机系统所使用的字符集
 - ASCII (美国标准信息交换码) 字符集
 - 规定了每个字符所对应的编码
- 一个字符以其对应的ASCII码的二进制形式存储在内存中
 - 一个字节，保存一个字符（英文字母、数字、符号）
- 字符常数就是一个普通整数，也可参与各种数学运算
 - 每个字符具有一个0~255之间的数值，可从ASCII表查出
 - 注意：'5'和整数5的区别

■ '5'的ASCII码值是53



2024/3/20

048	0	064	0	080	P	096	'	112	p
049	1	065	A	081	Q	097	a	113	q
050	2	066	B	082	R	098	b	114	r
051	3	067	C	083	S	099	c	115	s
052	4	068	D	084	T	100	d	116	t
053	5	069	E	085	U	101	e	117	u
054	6	070	F	086	V	102	f	118	v
055	7	071	G	087	W	103	g	119	w
056	8	072	H	088	X	104	h	120	x
057	9	073	I	089	Y	105	i	121	y
058	:	074	J	090	Z	106	j	122	z
059	;	075	K	091	[107	k	123	{
060	<	076	L	092	\	108	l	124	:
061	=	077	M	093	J	109	m	125	>
062	>	078	N	094	^	110	n	126	~
063	?	079	O	095	_	111	o	127	△

4.1 单个字符的输入/输出

- 通过调用标准库函数来实现

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

- 字符输出函数

- **putchar(ch)**
- 输出一个字符ch

- 字符输入函数

- **getchar()**
- 无参数
- 函数值为从输入设备接收的字符



【例4.1】大小写英文字母转换

```
1 #include <stdio.h>
2 main()
3 {
4     char ch;
5     printf("Press a key and then press Enter:");
6     ch = getchar();
7     ch = ch + 32; /* 将大写英文字母转换为小写英文字母 */
8     putchar(ch); /* 在屏幕上显示变量ch中的字符 */
9     putchar('\n'); /* 输出一个回车换行控制符 */
10 }
```

Press a key and then press Enter: B↙

b

What does *this statement* mean?

4.2 数据的格式化屏幕输出

■ 格式

printf(格式控制字符串, 输出项表列) ;

printf ("a=%d b=%f" , a , b) ;

函数名

普通
字符

格式
说明

可选输
出表列

■ 可以输出若干任意类型的数据

printf() 格式字符

%c character 以字符形式输出单个字符

%s string 输出一个字符串

%d decimal 以带符号十进制整数输出

%f float 以小数形式输出浮点数（6位小数）

%e exponent 以标准指数形式输出（6位小数）

%g 选用**%f**, **%e**中输出宽度较小的一种格式

%o octal 以八进制无符号整数输出（无前导0）

%x hex 以十六进制无符号整数输出（无前导0x）

%u unsigned 以十进制无符号整数输出

【例4.2】大小写英文字母转换

```
1 #include <stdio.h>
2 main()
3 {
4     char ch;
5     printf("Press a key and then press Enter:");
6     ch = getchar();
7     ch = ch + 32;
8     printf("%c, %d\n", ch, ch);
9 }
```

Press a key and then press Enter: B↙
b, 98

printf() 的格式修饰符

- m** 表示数据占用的最小宽度
数据宽度大于**m**, 按实际宽度输出
数据宽度小于**m**时, 补空格
- n** 对实数表示输出**n**位小数
对字符串表示最多输出的字符个数
- l** 长整型整数, 加在**d**、**o**、**x**、**u**前
- L** **long double**型数, 加在**f**、**e**、**g**前
- 改变输出内容的对齐方式
默认为右对齐

printf() 的格式修饰符

- Format for float: **%f**

- General format:

%[<最小域宽>. <小数位数>]f

Example:

```
printf('Value is: %10.4f', 32.6784728);
```

4 digits

Value is: 32.6785

10 characters

printf() 的格式修饰符

Example:

```
printf('Value is: %10f', 32.6784728);
```

6 digits (default)

Value is: | 32.678473 |

10 characters

printf() 的格式修饰符

Example:

```
printf('Value is: %.3f', 32.6784728);
```

3 digits



```
Value is:32.678
```

【例4.3】计算圆的周长和面积

```
#include <stdio.h>
main()
{
    const double pi = 3.14159;
    double r, circum, area;
    printf("Input r:");
    scanf("%lf", &r);
    circum = 2 * pi * r;
    area = pi * r * r;
    printf("printf WITHOUT width or precision specifications:\n");
    printf("circumference = %f, area = %f\n", circum, area);
    printf("printf WITH width and precision specifications:\n");
    printf("circumference = %7.2f, area = %7.2f\n", circum, area);
}
```

Input r: 5.3↙

printf WITHOUT width or precision specifications:

circumference = 33.300854, area = 88.247263

printf WITH width and precision specifications:

circumference = 33.30, area = 88.25

4.3 数据的格式化键盘输入

■ 格式

scanf(格式控制字符串, 地址表列);

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

格式字符
指定输入数
据格式

非格式字符
输入数据以, 分隔
没有时可以空格、
Tab或回车分隔

输入数据
地址表列

scanf() 的格式字符

- c** 以字符形式输入单个字符
- s** 输入字符串，以非空字符开始，遇第一个空白字符结束
- d** 以带符号十进制形式输入整型数据
- f** 以小数形式输入浮点数
- e** 以标准指数形式输入
- o** 以八进制无符号形式输入（无前导0）
- x** 以十六进制无符号形式输入（无前导0x）

*scanf()*的格式修饰符

- m** 表示数据占用的宽度
- l** 加在**d**、**o**、**x**、**u**前：输入长整型
加在**f**、**e** 前：输入双精度型
- L** 加在**f**、**e** 前：输入**long double**型
- h** 加在**d**、**o**、**x** 前：输入短整型
- *** 本输入项在读入后不赋给相应的变量

输入数据的格式控制

```
#include <stdio.h>
main()
{
    int a, b;
    printf("Please input a and b:");
    scanf('%2d %2d %2d', &a, &b);
    printf("a=%d, b=%d, a+b=%d\n", a, b, a+b);
}
```

跳过一个输入项

Please input a and b: 123456↙

a=12, b=56, a+b = 68

输入数据的格式控制——例4.4

```
#include <stdio.h>
main()
{
    int a, b;

    scanf("%d %d", &a, &b);
    printf("a = %d, b = %d\n", a, b);
}
```

12 34↙

问题1：当要求程序输出结果为

a = 12, b = 34

时，用户应该如何输入数据？

输入数据的格式控制——例4.4

```
#include <stdio.h>
main()
{
    int a, b;

    scanf("%d,%d", &a, &b);
    printf("a = %d, b = %d\n", a, b);
}
```

问题2：当限定用户输入数据以逗号为分隔符，即输入数据格式为：
12,34↙
时，应修改程序中的哪条语句？怎样修改？

输入数据的格式控制——例4.4

```
#include <stdio.h>
main()
{
    int a, b;
    scanf("%d %d", &a, &b);
    printf("a = %d, b = %d\n", a, b);
}
```

a = 12, b = 34↙

问题3：语句`scanf("%d %d", &a, &b);`修改为
`scanf("a = %d, b = %d", &a, &b);`时，用户应该如何输入数据？

输入数据的格式控制——例4.4

```
#include <stdio.h>
main()
{
    int a, b;

    scanf(" %2d%2d ", &a, &b);
    printf("a = %d, b = %d\n", a, b);
}
```

问题4：限定用户输入数据为以下格式

1234↙

同时要求程序输出结果为a = 12, b = 34

输入数据的格式控制——例4.4

```
#include <stdio.h>
main()
{
    int a, b;

    scanf("%d %d", &a, &b);
    printf("a = |\\"%d\\"| ,b = |\\"%d\\"| \n", a, b);
}
```

问题5：限定用户输入数据为以下格式

12↙

34↙

同时要求程序输出结果为 a = "12" ,b = "34"

输入数据的格式控制——例4.4

```
#include <stdio.h>
main()
{
    int a, b;

    scanf ("%d%c%d", &a, &b);
    printf ("a = %d, b = %d\n", a, b);
}
```

问题6：设计程序使得用户可以以任意字符（回车、空格、制表符、逗号、其他）作为分隔符进行数据的输入

输入数据的格式控制——例4.4

```
#include <stdio.h>
main()
{
    int a, b;

    scanf ("%d%*2d%d", &a, &b);
    printf ("a = %d, b = %d\n", a, b);
}
```

问题7： 输入123456时程序的输出结果是什么？

a = 12, b = 56

输入数据的格式控制——例4.4

```
#include <stdio.h>
main()
{
    int a, b;

    scanf("%d %d", &a, &b);
    printf("a = %d, b = %d\n", a, b);
}
```

问题8：如果用户输入了非法字符，例如输入了12 3a，那么程序运行结果如何？

```
12 3a↙
a = 12, b = 3
```

输入数据的格式控制——例4.4

```
#include <stdio.h>
main()
{
    int a, b;

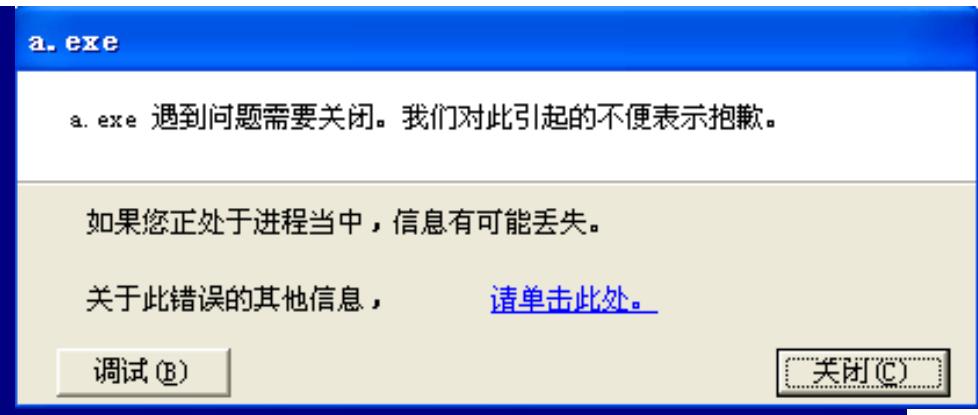
    scanf("%d %d", &a, &b);
    printf("a = %d, b = %d\n", a, b);
}
```

问题9：如果用户输入的是123a，那么结果又会如何呢？

```
123a↙
a = 123, b = -858993460
```

输入数据的格式控制——例4.4

```
#include <stdio.h>
main()
{
    int a, b;
    scanf("%d %d", &a, &b);
    printf("a = %d, b = %d\n", a, b);
}
```



问题10：如果程序第5行语句修改为...，那么结果又会如何呢？

```
warning: local variable 'b' used without having been initialized
warning: local variable 'a' used without having been initialized
```

C格式符的问题及解决—例4.5

```
#include <stdio.h>
main()
{
    int      data1, data2, sum;
    char    op;

    printf("Please enter the expression
            data1 + data2\n");
    scanf('%d%c%d', &data1, &op, &data2);
    printf("%d%c%d = %d\n",
           data1, op, data2, data1+data2);
}
```

第1次测试

Please enter the expression data1 + data2

12 | + 3 ↴

12 4199288 = 4199300

C格式符的问题及解决—例4.5

```
#include <stdio.h>
main()
{
    int      data1, data2, sum;
    char    op;

    printf("Please enter the expression
            data1 + data2\n");
    scanf('%d%c%d', &data1, &op, &data2);
    printf("%d%c%d = %d\n",
           data1, op, data2, data1+data2);
}
```

第2次测试 Please enter the expression data1 + data2

12 | 3 ↴

12 3 = 15

C格式符的问题及解决—例4.5

```
#include <stdio.h>
main()
{
    int      data1, data2, sum;
    char    op;

    printf("Please enter the expression
            data1 + data2\n");
    scanf('%d%c%d', &data1, &op, &data2);
    printf("%d%c%d = %d\n",
           data1, op, data2, data1+data2);
}
```

第3次测试 Please enter the expression data1 + data2

12|3↙

12+3 = 15

C格式符的问题及解决—例4.6

```
#include <stdio.h>
main()
{
    int      a;
    char     b;
    float   c;
    printf("Please input an integer: ");
    scanf("%d", &a);
    printf("integer: %d\n", a);
    printf("Please input a character: ");
    scanf("%c", &b);
    printf("character: %c\n", b);
    printf("Please input a float number: ");
    scanf("%f", &c);
    printf("float: %f\n", c);
}
```

希望得到的运行结果

```
Please input an integer: 12
integer:12

Please input an character : a
character :a

Please input a float number: 3.5
float number:3.500000
```

C格式符的问题及解决—例4.6

```
#include <stdio.h>
main()
{
    int      a;
    char     b;
    float   c;
    printf("Please input an integer:");
    scanf("%d", &a);
    printf("integer: %d\n", a);
    printf("Please input a character:");
    scanf("%c", &b);
    printf("character: %c\n", b);
    printf("Please input a float number:");
    scanf("%f", &c);
    printf("float: %f\n", c);
}
```

结果好像很奇怪呀！

```
Please input an integer: 12
integer:12
Please input an character :character
Please input a float number: 3.5
float number:3.500000
```

C格式符的问题及解决—例4.6

```
#include <stdio.h>
main()
{
    int      a;
    char     b;
    float   c;
    → printf("Please input an integer: ");
    → scanf("%d", &a);
    → printf("integer: %d\n float number: %f\n", a, 3.5);
    → printf("Please input a character: ");
    → getchar(); /*将存于缓冲区中的回车字符读入，避免被后面的变量作为有效字符读入*/
    → scanf("%c", &b);
    → printf("character: %c\n", b);
    → printf("Please input a float number: ");
    → scanf("%f", &c);
    → printf("float: %f\n", c);
}
```

程序修改后得到的运行结果

Please input an integer: 12↙
integer:12
Please input an character : a↙
character :a

Please input a float number: 3.5↙

float number:3.500000

C格式符的问题及解决—例4.6

```
#include <stdio.h>
main()
{
    int      a;
    char     b;
    float   c;
    printf("Please input an integer:");
    scanf("%d", &a);
    printf("integer: %d\n", a);
    printf("Please input a character:");
    scanf(' %c', &b); /*第2种解决方案:在%c前加一个空格*/
    printf("character: %c\n", b);
    printf("Please input a float number:");
    scanf("%f", &c);
    printf("float: %f\n", c);
}
```

再回头来看例4.5 如何以任意分隔符输入加法算式？

12+3↙

12 + 3↙

12↙
+↙
3↙

```
#include <stdio.h>
main()
{
    int      data1, data2, sum;
    char    op;

    printf("Please enter the expression
            data1 + data2\n");
    scanf("%d %c%d", &data1, &op, &data2);
    printf("%d%c%d = %d\n",
           data1, op, data2, data1+data2);
}
```

格式输入函数常见错误

- `scanf ("%d,%f\n", &a, &b);`
- `scanf ("%d,%f", a,b);`
- `scanf ("%7.2f", &a);`

■ 几点忠告

- 不要拘泥于细节
- 不要死记硬背
- 在使用中慢慢掌握





■ Questions and answers

