



西北工业大学  
NORTHWESTERN POLYTECHNICAL UNIVERSITY

---

# C程序设计 Programming in C



**1011014**

---

主讲：姜学锋，计算机学院

## 编程实现简单数据的输入（1）

- ◆ 1、scanf函数
- ◆ 2、输入格式控制

### 3.2.3 格式化输入

---

- ▶ 1. scanf函数
- ▶ scanf函数的作用是从标准输入设备（键盘终端）读取格式化的数据，一般形式为：

```
scanf( 格式控制, 输入项列表..... );
```

- ▶ 例如：

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

### 3.2.3 格式化输入

---

- ▶ scanf函数将数据读到输入项列表中，每个输入项必须为地址形式（&变量）。

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

### 3.2.3 格式化输入

---

#### 格式输入详解

如果输入项不是地址形式（即没有 &），会导致严重错误。例如：

```
1 int a , b;  
2 scanf("%d%d", a, b); //导致严重错误
```

原因：scanf的原理是将输入的数据往“变量所在内存单元”中送，&a表示变量a所在内存单元的地址，往这个地址送就是往变量a中。而a表示变量a的值，这个值是“变量a所在内存单元的地址”的可能性很低。送到别的内存单元会导致“内存存取异常”错误。

### 3.2.3 格式化输入

---

- ▶ 格式控制可以包含下列情况的一种或多种：
- ▶ (1) 空白符：
  - ▶ ①空格（' '）；②TAB（'\t'）；③换行（'\n'）。
- ▶ 空白符使scanf读输入数据中的连续空白符，但不保存它们，直至读到下一个非空白字符为止。格式串中的一个空白字符可以匹配任意数目（包括0个）和任意组合的空白符。

### 3.2.3 格式化输入

---

#### 格式输入详解

例如：

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

如果输入数据 12  56↵，则a=12,b=56（↵表示按回车）

如果输入数据 12     56↵，则a=12,b=56

如果输入数据 12  →  → 56↵，则a=12,b=56（→表示按TAB）

如果输入数据 12  ↵  56↵，则a=12,b=56

### 3.2.3 格式化输入

---

#### 格式输入详解

例如：

```
1 int a;  
2 scanf("%d\n", &a);
```

如果输入数据 12↵，scanf不会结束，依然等待输入。

原因：写在scanf的三个空白符“空格、\t、换行\n”的任意数目、任意组合形式均会“吃掉”输入时给出的连续多个“空格、TAB、换行”任意数目、任意组合数据，这样scanf始终得不到“回车”，从而依然等待输入。



### 3.2.3 格式化输入

---

- ▶ (2) 除百分号 (%) 外的非空白符：非空白符使scanf读入一个匹配的非空白字符，但不保存它。如果输入中的字符并不匹配，则立即终止scanf。

### 3.2.3 格式化输入

---

#### 格式输入详解

```
1 int a=1, b=2;  
2 scanf("%d%d", &a, &b); //如果输入"34 56", 则a=34,b=56,格式要求  
   (空格、TAB、回车)与实际输入匹配  
3  
4 scanf("%d%d", &a, &b); //如果输入"34, 56", 则a=34,b=2,格式要求  
   (空格、TAB、回车)与实际输入(,)不匹配,因而尚未处理b时scanf就结束了。
```

### 3.2.3 格式化输入

---

#### 格式输入详解

```
1 int a=1, b=2;  
2 scanf("%d,%d", &a, &b); //如果输入"34 56", 则a=34,b=2,格式要求  
  (,) 与实际输入(空格)不匹配, 因而尚未处理b时scanf就结束了。  
3  
4 scanf("%d,%d", &a, &b); //如果输入"34, 56", 则a=34,b=56,格式要  
求(,) 与实际输入匹配  
5  
6 scanf("%d%d", &a, &b); //如果输入"34x86", 则a=34,b=2,格式要求  
  (空格、TAB、回车) 与实际输入(x)不匹配, 因而尚未处理b时scanf就结束了。
```

### 3.2.3 格式化输入

---

#### 格式输入详解

```
1 int a=1, b=2;  
2 scanf("a=%db=%d", &a, &b); //如果输入"34 56", 则a=1,b=2, 格式  
   要求(a=)与实际输入(空格)不匹配, 因而尚未处理a时scanf就结束了。  
3  
4 scanf("a=%db=%d", &a, &b); //如果输入"a=34b=56", 则a=34,b=56,  
   格式要求(a= b=)与实际输入匹配  
5  
6 scanf("a=%db=%d", &a, &b); //如果输入"a=34 b=56", 则a=34,b=2,  
   格式要求(b=)与实际输入(空格b=)不匹配, 因而尚未处理b时scanf就结束了。
```

### 3.2.3 格式化输入

---

#### 格式输入详解

```
1 int a=1, b=2;  
2 scanf("a=%d,b=%d", &a, &b); //如果输入"a=34,b=56", 则a=34,b=56, 格式要求(a= ,b=)与实际输入匹配  
3  
4 scanf("a=%d,b=%d", &a, &b); //如果输入"a=34b=56", 则a=34,b=56, 格式要求(,b=)与实际输入(b=)不匹配, 因而尚未处理b时scanf就结束了。
```

### 3.2.3 格式化输入

---

#### 格式输入详解

```
1 int a=1, b=2;  
2 scanf("Please input a=%d", &a); //需要输入"Please input a=XX  
X" (xxx表示整型数据) 时格式要求和实际输入才匹配, 任意一个字符 (含大小写)  
不同导致a得不到正确的输入。  
3  
4 printf("Please input a="); //输入之前先输出, 即给出输入提示  
5 scanf("%d", &a);
```

### 3.2.3 格式化输入

---

#### 格式输入详解

```
1 int a,b,c;  
2  
3 scanf("%d,%d,%d",&a,&b,&c); //输入必须匹配一般字符  
4 //输入: 1,2,3      结果a=1,b=2,c=3  
5 //输入: 1 2 3      结果a=1,b,c不确定 (输入空格不匹配逗号, scanf终止)  
6 scanf("a=%db=%dc=%d",&a,&b,&c); //输入必须匹配一般字符  
7 //输入: a=1b=2c=3    结果a=1,b=2,c=3  
8 //输入: 1 2 3      结果a,b,c不确定 (输入不匹配a=, scanf终止)
```

### 3.2.3 格式化输入

---

- ▶ (3) 格式说明符：由百分号%开头。格式说明符使scanf读入输入数据中的字符，并将它转换成指定类型的值，该值将保存到输入项中。



### 3.2.3 格式化输入

---

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

► 格式控制自左向右地处理。

- 遇到第1个格式说明符时，读入第1个输入数据，并存放到第1个输入项中，依此类推，直到格式控制串结束。
- 其他字符用来匹配输入中的字符序列，输入中的匹配字符被读入但不保存；如果输入中某个字符与格式说明相冲突，则scanf终止，该字符将留在输入流中，就象没有读过一样。

### 3.2.3 格式化输入

#### 格式输入详解

```
1 int a=7,b=8;  
2 scanf("%d%d", &a , &b); //如果输入“12 34↵”，则a=12、b=34，格式要求数目（2）与实际输入相同  
3 scanf("%d%d", &a , &b); //如果输入“12↵”，则会继续提示输入  
4 scanf("%d%d", &a , &b); //如果输入“12 34 56↵”，则a=12、b=34，多余的输入（56）留在缓冲区中，下次读时处理  
5 scanf("%d", &a , &b); //如果输入“12 34↵”，则a=12、b=8，格式要求数目（1）少于实际输入（2）  
6 scanf("%d%d%d", &a , &b); //如果输入“12 34 56↵”，则a=12、b=34，然后出现严重错误，格式要求数目（3）多于实际输入
```

### 3.2.3 格式化输入

---

- ▶ 输入数据被定义为：
- ▶ ① 下一个空白符之前的所有字符；
- ▶ ② 下一个不能按格式说明转换的字符之前的所有字符（如在八进制下出现8）；
- ▶ ③ 到达域宽之前的所有字符。

### 3.2.3 格式化输入

---

#### 格式输入详解

```
1 int a=7,b=8;  
2 scanf("%d%d", &a , &b); //如果输入“12 34↵”，则a=12、b=34，  
   格式要求（用空格、TAB、回车）作为2个输入数据间隔，与实际输入匹配  
3  
4 scanf("%d%d", &a , &b); //如果输入“12→34↵”，则a=12、b=34，格  
   式要求（用空格、TAB、回车）作为2个输入数据间隔，与实际输入匹配
```

### 3.2.3 格式化输入

---

#### 格式输入详解

```
1 int a=7,b=8;  
2 scanf("%d%d", &a , &b); //如果输入“12↵34↵”，则a=12、b=34，格式  
   要求（用空格、TAB、回车）作为2个输入数据间隔，与实际输入匹配  
3  
4 scanf("%o%o", &a , &b); //如果输入“56 78↵”，则a=46、b=7，格式  
   要求（八进制）作为输入数据与实际输入（56）匹配，但第2个数据（八进制）不  
   识别“8”终止
```

### 3.2.3 格式化输入

---

举例：

```
1 int a,b,c;  
2  
3 scanf("%d%d%d",&a,&b,&c); //连续输入用空格、TAB、回车间隔  
4 //输入: 1 2 3      结果a=1,b=2,c=3  
5 //输入: 1,2,3      结果a=1,b,c不确定（输入逗号不匹配空白符，scanf终止）
```

### 3.2.3 格式化输入

---

- ▶ 如果输入项个数比给定的格式说明符多，多余的输入项被忽略；
- ▶ 如果输入项个数比格式说明符少，则读入结果是不可预料的。
- ▶ scanf函数经常会引发灾难性的结果，其原因就是格式说明、输入项、实际输入数据不匹配。

### 3.2.3 格式化输入

---

举例：

```
1 int a,b,c;  
2  
3 scanf("%d%d",&a,&b,&c); //格式数目小于输入项数，多余输入项未被输入  
4 scanf("%d%d%d",&a,&b); //格式数目大于输入项数，崩溃性错误
```



**CP 程序设计**