

第3章 键盘输入与屏幕输出

——数据的格式化屏幕输出

本节要讨论的主要问题

- C语言如何实现数据的输入和输出？
- 如何向屏幕输出一串字符信息？
- 如何在屏幕上显示变量的值？



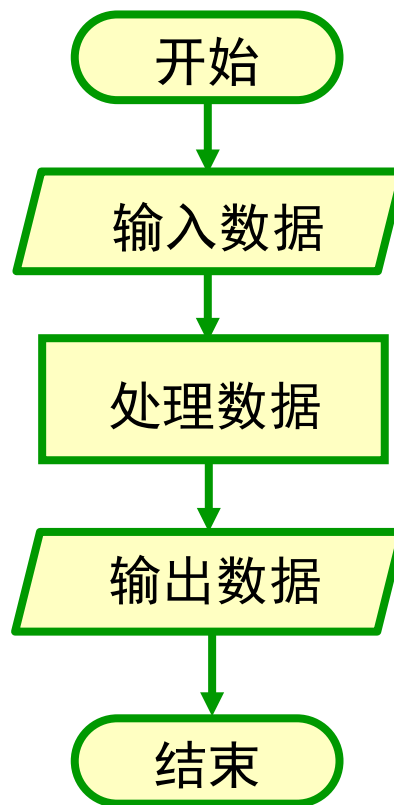
顺序结构

- 一般而言，顺序结构程序涉及如下三个基本操作：

- * 输入数据
- * 处理数据
- * 输出数据

- 顺序结构的特点

- * 自上而下，依次按顺序执行



C语言如何实现数据的输入和输出?

- 程序获得输入数据的方式: **键盘**, 文件
- 程序输出数据的方式: **屏幕**, 文件

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

```
int main()
```

```
{
```

```
    scanf(".....");
```

```
    .....
```

```
    printf("Hello!\n");
```

```
    .....
```

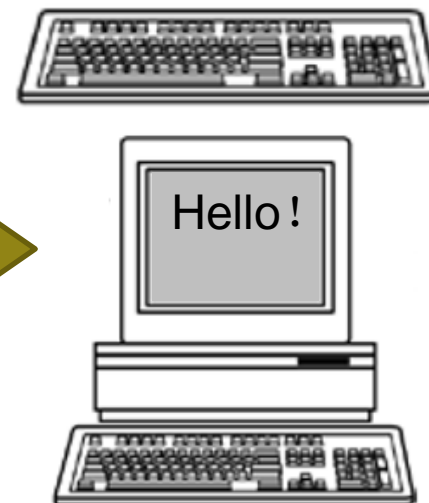
```
    return 0;
```

```
}
```

编译预处理命令: 在编译之前进行的处理

◀ getting keyboard input

displaying output on screen ▶



数据的格式化屏幕输出

- 按指定格式和类型输出变量的值，或者输出一行字符串

printf (格式控制字符串, 输出值参数表);

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

普通
字符

转换说明

指定输出
数据格式

可变长度的
输出参数表

与格式说明
符一一对应

- * 输出值参数类型应与格式转换说明符相匹配
- * 可输出多个任意类型的数据

printf()格式字符

- %d** decimal 输出十进制有符号int型
- %u** unsigned 输出十进制无符号int型
- %f** float 以小数形式(6位小数)输出float,double型
- %e** exponent 以标准指数形式输出float,double型

```
int a = 10;
```

```
printf("%d", a);
```

10

```
double b = 10.3;
```

```
printf("%f", b);
```

10.300000

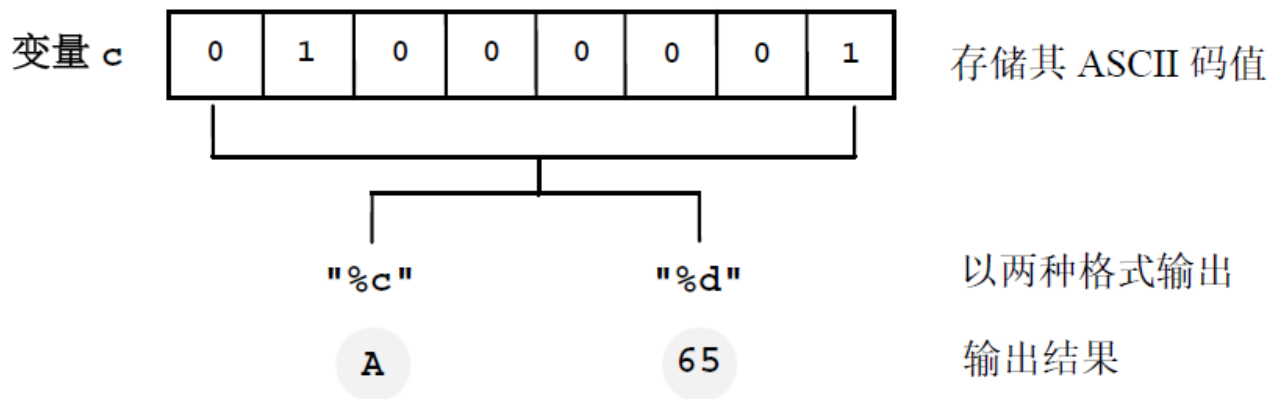
```
float b = 10.3;
```

```
printf("%e", b);
```

1.030000e+001

printf()格式字符

%c character 以字符形式输出单个字符



```
char c = 'A';
```

```
printf("%c", c);
```

A

```
char c = 'A';
```

```
printf("%d", c);
```

65

printf()的格式修饰符

- l 加d、u前输出long型
- h 加d、u前输出short型

```
long a = 10;
```

```
printf("%ld", a);
```

```
10
```

```
short b = 10;
```

```
printf("%hd", b);
```

```
10
```


printf()的格式修饰符

- m** 输出数据的**最小域宽**
数据位宽大于m，按实际位宽输出
数据位宽小于m时，**右对齐，左补空格**

Example:

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

6 digits (default)



Value is: || 32.678473

10 characters右对齐

printf()的格式修饰符

m 数据占用的最小域宽

数据位宽小于m时，右对齐，左补空格

-m 左对齐，右补空格

Example:

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

6 digits (default)

Value is: 32.678473

10 characters 左对齐

printf()的格式修饰符

显示精度.n 对浮点数表示输出n位小数

格式: %.nf

Example:

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

3 digits



Value is 32.678

printf()的格式修饰符

显示精度.n 对浮点数表示输出n位小数

一般格式: %m.nf

Example:

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

4 digits

Value is: 32.6785

10 characters

printf()的格式修饰符

显示精度.n 对浮点数表示输出n位小数

一般格式: %m.nf

Example:

```
printf("Value is:%10.0f", 32.6784728);
```

Value is:

33

10 characters

printf()格式字符

- 问题：printf()用%作为格式字符的标识，那么如何输出%呢？

```
int a = 10;  
printf("%d%%", a);
```

10 %

