printf的格式控制的完整格式：  
% - 0 m.n l或h 格式字符  
下面对组成格式说明的各项加以说明：  
①%：表示格式说明的起始符号，不可缺少。  
**②-：有-表示左对齐输出，如省略表示右对齐输出。**  
**③0：有0表示指定空位填0,如省略表示指定空位不填。**  
**④m.n：m指域宽，即对应的输出项在输出设备上所占的字符数。N指精度。用于说明输出的实型数的小数位数。为指定n时，隐含的精度为n=6位。**  
⑤l或h:l对整型指long型，L(大写)对实型指long double型(TC下)。h用于将整型的格式字符修正为short型。

**格式字符**  
格式字符用以指定输出项的数据类型和输出格式。  
①d格式：用来输出十进制整数。有以下几种用法：  
%d：按整型数据的实际长度输出。  
**%md：m为指定的输出字段的宽度。如果数据的位数小于m，则左端补以空格，若大于m，则按实际位数输出。  
%ld：输出长整型数据**。  
②o格式：以无符号八进制形式输出整数。对长整型可以用"%lo"格式输出。同样也可以指定字段宽度用“%mo”格式输出。  
例：  
main()  
{ int a = -1;  
printf("%d, %o", a, a);  
}  
运行结果：-1,177777  
程序解析：-1在内存单元中（以补码形式存放）为(1111111111111111)2，转换为八进制数为(177777)8。  
③x格式：以无符号十六进制形式输出整数。对长整型可以用"%lx"格式输出。同样也可以指定字段宽度用"%mx"格式输出。  
④u格式：以无符号十进制形式输出整数。对长整型可以用"%lu"格式输出。同样也可以指定字段宽度用“%mu”格式输出。  
⑤c格式：输出一个字符。  
⑥s格式：用来输出一个串。有几中用法  
%s：例如:printf("%s", "CHINA")输出"CHINA"字符串（不包括双引号）。  
%ms：输出的字符串占m列，如字符串本身长度大于m，则突破获m的限制,将字符串全部输出。若串长小于m，则左补空格。  
%-ms：如果串长小于m，则在m列范围内，字符串向左靠，右补空格。  
**%m.ns：输出占m列，但只取字符串中左端n个字符。这n个字符输出在m列的右侧，左补空格。**  
%-m.ns：其中m、n含义同上，n个字符输出在m列范围的左侧，右补空格。如果n>m，则自动取n值，即保证n个字符正常输出。  
⑦f格式：用来输出实数（包括单、双精度），以小数形式输出。有以下几种用法：  
%f：不指定宽度，整数部分全部输出并输出6位小数。  
**%m.nf：输出共占m列，其中有n位小数，如数值宽度小于m左端补空格。**  
%-m.nf：输出共占n列，其中有n位小数，如数值宽度小于m右端补空格。  
⑧e格式：以指数形式输出实数。可用以下形式：  
%e：数字部分（又称尾数）输出6位小数，指数部分占5位或4位。  
%m.ne和%-m.ne：m、n和”-”字符含义与前相同。此处n指数据的数字部分的小数位数，m表示整个输出数据所占的宽度。  
⑨g格式：自动选f格式或e格式中较短的一种输出，且不输出无意义的零。

关于printf函数的进一步说明：  
如果想输出字符"%",则应该在“格式控制”字符串中用连续两个%表示，如:  
printf("%f%%", 1.0/3);  
输出0.333333%。

**对于单精度数，使用%f格式符输出时，仅前7位是有效数字，小数6位．  
对于双精度数，使用%lf格式符输出时，前16位是有效数字，小数6位．**