没讲到的补充：

程序中所有的符号都应该是英文状态下的符号，比如每一条指令最后的分号; 写成；（中文分号编译器会报错，就是有带颜色下划线，并且想要运行也运行不了——编译不通过）

还有像括号（），引号‘’ “”都要是英文状态下的

程序的注释，解释代码的意思 在代码后面使用两个斜杠// ，比如

C=a+b;//把a和b的值加起来，结果赋给c

#include <stdio.h> 函数printf scanf

2^10 pow(2,10) <math.h>

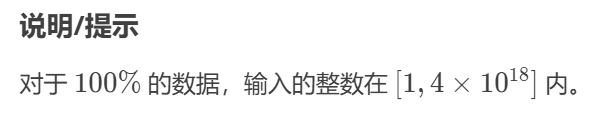
功能库 代码库 函数库

int -2,147,483,648 至 2,147,483,647 2\*10^9 一般都够用了

如果题目里面数字过大，要开long int 或者long long int

记忆范围用处：

做题时要看数据的范围，选择合适的类型



这种要选择long long int

Main函数 主函数 只有一个{ }

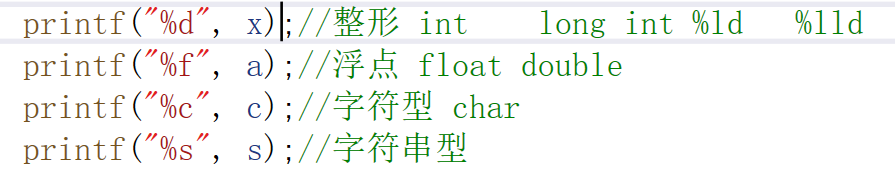
Void 空

Float 浮点类型 float 叫单精度浮点数 Double是双精度浮点数

Char 字符型 a b c % ^ + = \_ 1 2 3 （ASCII码 之后再学） 单引号

Char\* 字符串 helloworld 双引号

输出 printf 占位符 格式printf(“”,\_\_\_);



这里纠正一下，double对应的占位符是%lf

\n 换行

%10d控制最短长度 如果12位数，还是显示12位

%010d 补0 %-10d 顶着左端

%10.4f 4位小数 占10位 四舍五入

&x x的地址 scanf("%d", &x);

整数除法 实质是向下取整 比如 3/2=1 5/3=1

z % y得到余数

要想得到精确值：

1. 用一个浮点数来存储结果②强制类型转换：

Int a=3;

Int b=2;

Float ans；

Ans= (float)a/b;

或者 ans=a/(float)b

或者ans=(float)a/(float)b

但是不能是 ans=(float)(a/b)

本质：两个运算数中至少要有一个是浮点数，使得计算机认为这个除法运算是浮点数之间的运算而不是整数的运算

从原理方面，还有一种更改方案：

float a=3;

Int b=2;

Float ans；

Ans= a/b;

一般不用，这种属于修改语义，比如3个苹果分给2个人，那么苹果数x应该是整数，改成浮点数会影响程序可读性