

- 前言
- scanf()和printf()的使用
 - 函数原型
 - 常见错误
 - "&"
 - 引号位置
 - "EOF"
- 分号
 - 应该出现在哪里
 - 常见错误
 - 乱打分号
 - 用中文分号
- if
 - == 而不是 =
 - 不要在 if() 里写超长一串
- ide使用
 - 重新编译
 - 字符画
- 变量类型
 - int 和 long long
 - char
 - 数据范围
 - 类型转换

前言

对于很多同学，航C（即C语言程序设计）这门课是上的第一门编程课——或者你之前接触过一点 Python，however——C语言的难度并不低。课堂上老师强调的多是语法学习，类比到自然语言的的学习，就是教你一句话怎么说单词怎么用，而编程尤其是航C包含的题，则是要你写一篇能发表的作文出来。

我的看法是：不要着急和最厉害的那些同学比，一点点来，形成良好的代码习惯不要像我，多交流多讨论，积极写题，也不要过于较劲我觉得数分更重要。参考书倒不是必须的，虽然现在 CSDN 风评不太好，不过上一上找找别人的代码思路还是很不错的。善用搜索引擎！！你的大部分问题别人都问过了。有余力多看看函数原型。

航C加油站系列会包括 错误分析 和 知识点补充 两个部分，希望我可以把课堂和练习之间的距离缩小一点，然而我水平有限，喜欢的方向也不够全面，大概率做得不会太好，哈哈。

scanf()和printf()的使用

函数原型

```
int scanf(const char *format, ...); //scanf("<格式化字符串>", <参量表>);  
int printf(const char *format, ...); //printf("<格式化字符串>", <参量表>);
```

格式化输入输出的占位符号都可以查到。

常见错误

"&"

这个符号涉及到了指针的内容，现在我们只需要记住，scanf 读取（除字符串外）需要 &，printf 输出不需要。

```
//一个错误示范  
int a,b;  
scanf("%d%d", a,b);  
//另一个错误示范  
char s[100];  
scanf("%s", &s);
```

引号位置

只把输入输出的部分括起来。

```
//错误示范  
scanf("%d%d, &a, &n");
```

"EOF"

scanf 读到文件尾巴的时候会返回 EOF，在航C中常被用来停止读取循环：

```
while(scanf("%d", &a) != EOF){  
    //your code  
}
```

不过我并不喜欢出这种题我不会用。

同样的，也可以在读取到的个数（返回值）和所需要的数据数不同时停止循环：

```
while( scanf( "%d%d", &a, &b) != 2 ){  
    //your code  
}
```

分号

应该出现在哪里

应该明确的是，C语言中的分号对应自然语言中的句号，因此你每说一句话就应该打一个；，一句话没说完也不要打分号。{}括起来的内容相当于一句话，即代码块，其中每一句都应该打上分号。

常见错误

乱打分号

```
while(1);{  
    //your code  
}
```

以上面这段代码为例，while后紧跟分号显然是不对的，while是判断语句，在括号内成立时执行后面一行代码或代码块的内容，并不是一句完整的话，应该删去这个分号。

用中文分号

其实不仅是分号有这个问题，所有符号都有可能被打成全角符号，这样当然无法编译（得到一个CE）。我的建议是用ide自带的括号匹配检查打没打对。以及千万不要直接在oj提交栏里直接写代码，除非您已经超强了。

if

== 而不是 =

一个等号是赋值，怎样都是1，除非没赋值成功。
这一句请印在你的脑子里/

不要在 if() 里写超长一串

除非你的逻辑已经超强了，否则很容易出错。建议用if、else if和else判断。希望大家都有良好的代码习惯。

ide使用

重新编译

修改完代码记得点一下重新编译再运行，不要再问学姐学姐我怎么改了还是错的，我会和你一起找还找不出来，像两个笨蛋。

however,我使用的 vscode 并没有这个问题。

字符画

用查找替换功能把所有可疑分子(\%"')换掉，再合理使用 printf 就行。

变量类型

int 和 long long

短整数和长整数。还有无符号 unsigned 版本。不同编译器位数不同，自己试试。

char

八位的特殊整型。

北航的教材后有 ASCII 表，常拿出来对照看看。

数据范围

所有类型使用的时候都要防止溢出。当你输出不了合理答案的时候回头看一看题目的数据范围。之后会讲字符串模拟的操作

类型转换

还是要规范使用各种数据类型啊，每个版本的编译器都不太一样。

强制类型转换：

```
//例子
double f=114.514;
int a=(int)f;
```