本文档记录CPrimerPlus一书中所提到的C语言编程注意事项

1. main函数的标准形式（P18）

int main（）

{

return 0；

}

这种形式便于移植

1. 变量的声明位置（P19）

变量的声明位置应该尽量位于一个代码块的开始。以便于移植到老的机器上也是可行的

1. 变量名字的选择（P20）

变量名字第一个字符必须是下划线或者字母，变量名字可以由下划线和字母以及数字共同组成。

由于操作系统和C库通常采用以一个或两个下划线开始的名字。故自定义变量名尽量不使用下划线作为开始

1. 初始化变量（P37）

初始化变量时

int cows = 29，goats = 14； 这种是可以允许的

int cows， goats = 14； 这种做法不好，容易使人误以为cows也被初始化。

1. 多种整数类型选择（P39）

诸多整数类型中优先选择unsigned类型，没有负数且表示数值范围大

1. 常量（P41、P43、P48）

整型常量：初始化时未指定存储类型时，系统自动为其寻找能满足其大小的存储空间，浪费时间。初始化常量时可自行指定。

如 1000000L系统将自动用long int型的空间去存储该常量。（应该用L而不是l，l容易与数字1搞混）

再比如 2345L 表示用long int的空间去存储2345这个数，而不是按照int去存储。

浮点常量：默认情况，系统将浮点常量当做double存储。

如 some 是个float类型，some = 2.0 \* 4.0；

由于2.0和4.0均采用double类型存储，均为64位，乘积运算采用双精度，结果被截成正常的float 长度。虽然保证了计算的精度，但是浪费了计算时间。

也可采用整型常量的做法，3.45F即使用float类型去存储3.45这个常量。3.45L则表示按照long double类型存储

字符常量：C语言将字符常量视为int类型而非char类型。

如 char grade = ‘B’；

‘B‘作为数值66存储在一个int里面，而赋值后的grade则把66存储在一个8位单元中。

利用此特性可以定义一个字符常量‘FATE’，这是将四个独立的8位ASCII存储在一个32位单元中。如果把这个字符常量赋给一个char变量，那么只有最后8位会起作用，因此变量的值为‘E’。

1. printf的注意事项（P42）

printf（“双引号中有几个格式说明符，后面的参数就得有几个”）参数数量匹配

printf（“引号内部格式说明符表示的类型得和后面的参数一致”）格式一致

如uint16\_t类型的变量，打印的时候格式说明符得是能打印16位的，不能全部使用%d

1. 隐式格式转换（P42）

在short和int类型长度不一致的系统，C在处理short时会自动将short类型转换成int类型。int类型时计算机处理起来最方便最有效的整数类型。

9、可移植类型（P46）

头文件inttype.h中包含有可移植类型的变量别名

如int16\_t 、int32\_t、等

同样包含printf中格式说明符的可移植类型

10、浮点常量（P48）

没有小数点的数不一定不是浮点数

如2E5，使用这种形式就表示其为浮点数

浮点常量可以省略小数点可以省略指数部分，这两者不能同时省略

可以省去小数部分也可以省略整数部分，这两者也不能同时省略。

11、printf的使用（P53）

printf（“adada”“asdwad”“asdwa”）；可以将多个双引号中内容连接在一起打印出来

printf（“只要不在引号内部或者一个单词中间进行断行，使用就没有问题”）

12、