

程序代码编写规范

2019.9

【功能设计】

(1) 尽可能消除编译中的错误和警告信息。

(2) 可靠性要求，程序运行时，输入数据有可能超界、文件打开可能失败、数据数量有可能太多或太少，任何情况下，尽可能使程序稳定运行，不能导致程序崩溃或死机。

(3) 在可能情况下（与运行示例不冲突），做到界面交互友好，输入之前有提示，输入之后有反馈，异常情况有信息等。

【代码风格】

(1) 代码布局，程序各部分代码，按以下顺序排列：

- ① 包含语句集中在代码最前端，库函数使用尖括号，如：`#include <stdio.h>`
- ② 专题另行提供的代码资源，头文件使用双引号，如：`#include "mylib.h"`
- ③ 预定义语句，预定义的内容和每个参数都加上括号，如：`#define PRICE (0.6)`
- ④ 结构体、枚举等全局性数据类型的声明。
- ⑤ 全局变量及其初始化，全局函数声明（如果函数实现自底向上排列，函数声明可省略）。
- ⑥ 各函数的实现（建议自底向上排列）
- ⑦ 主函数，除明确要求命令行参数外，主函数统一使用 `int main(void);`

(2) 变量与函数命名

- ① 预定义、自定义函数、结构体类型、全局变量、主要局部变量等，必须使用有意义的函数名称，并注释说明其功能/作用。
- ② 除 `i, j` 等简单变量外，其他有意义的变量尽可能不要重名（即使在不同函数中）。

(3) 程序缩格

- ① 程序代码的层次清晰，缩格使用 4 个空格。
- ② 复合语句的“{”和“}”，单独占用一行。
- ③ 一般情况下，每行一条语句（相邻行上下对称布局除外）。

(4) 保持程序舒展

- ① 赋值运算符两边至少一个空格。
- ② `if/for/while/switch` 与左括号之间至少一个空格。
- ③ 函数参数之间（逗号之后）至少一个空格。
- ④ 表达式各主要项之间空格隔离。

【程序注释】

(1) 程序的第一行应为注释，包括程序名称、程序功能、编写日期等。

(2) 预定义、自定义函数、结构体类型及各成员、全局变量、主要局部变量等，注释说明。

(3) 函数中的主要代码段，注释说明该代码段的功能及设计思路。

【代码范本】

```

1 //Example.c, 2019年, 程序代码示例
2 #include <stdio.h> //库函数, 标准输入输出
3 #include "mylib.h" //外部资源库
4 #define max(a,b) ((a)>(b)?(a):(b)) //每个参数加括号、整体加括号
5 ...//全局结构体类型、typedef等, 各成员分别说明
6 ...//全局变量及其初始化(除非特别要求, 尽可能不使用全局变量)
7 ...//函数声明及其说明(如果函数自底向上排列, 可省略)
8 //各函数实现(建议按自底向上顺序)
9 ... func(.....)
10 { //函数func功能说明
11 } //函数func
12 int main(void)
13 { //主函数
14     //变量声明集中在函数前部, 关键变量说明其功能
15     for (i=0; i<n; i++) //代码舒展, 插入必要的空格
16         x = x + i; //下一层次代码, 缩进4个空格
17     return ...; //返回...
18 } //大括号占一行
19 //其他说明、运行示例等
20

```