1. 缩进

规则1-1：禁止使用TAB键，使用4个空格代替。

规则1-2：在switch语句中，每一个case和default要比switch缩进4个空格。

规则1-3：不要将多个语句放在同一行。

规则1-4：不要将多个赋值放在同一行（内核编程范式超简单，避免复杂的表达式）。

1. 换行

规则2-1：一行代码或注释不能超过120列（字符）。

规则2-2：长表达式（>120列）要在低优先级操作符处拆分成新行，操作符放在新行之首（以便突出操作符）。拆分出的新行要进行适当的缩进，使排版整齐。

规则2-3：以上规则对较长参数表的函数声明同样有效。

1. 大括号

规则3-1：将左大括号放置在行末尾，而右大括号放置在行首。

*if (true) {*

*do thing;*

*}*

规则3-2：对非函数的语句块也是如此（if、switch、for、while、do）。

*switch (action) {*

*case KEY\_UP:*

*return "up";*

*case KEY\_DOWN:*

*return "down";*

*default:*

*return NULL;*

*}*

规则3-3：函数的左大括号在新行的行首。

*int function(int x)*

*{*

*Function body;*

*}*

规则3-4：右大括号单独一行，除非后面跟着的是同一语句的继续。例如 do 语句中的 “while”或 “if ”语句中的 “else” 。

*do {*

*body of do-loop*

*} while (condition);*

和

i*f (x == y) {*

*..*

*} else if (x > y) {*

*...*

*} else {*

*....*

*}*

规则3-5：只有一行执行语句时也要加大括号。

*if (condition) {*

*action();*

*}*

规则3-6：判断语句中只有一个分支为单行语句时，不支持这样用，你需要在所有分支中都加入大括号。

*if (condition) {*

*do\_this();*

*do\_that();*

*} else {*

*otherwise();*

*}*

规则3-7：循环语句包含多条语句时，需要加入大括号。

*while (condition) {*

*if (test) {*

*do\_something();*

*}*

*}*

1. 空格

规则4-1：大多数函数在其关键字后面会加上空格。

规则4-2：sizeof, typeof, alignof, 和 \_\_attribute\_\_等像这样的长得像函数的，就不需要加空格。

*s = sizeof(struct file); √*

*s = sizeof( struct file ); ×*

规则4-3：在如下关键字if, switch, case, for, do, while后面加空格。

规则4-4：当声明指针或者声明一个返回指针的函数时，最好是把小小的“\*”离着标识符（不论是变量常量还是函数）近一点，而不是和数据类型近一点。

规则4-5：在单目运算符左边和右边加上空格( =、+、-、<、>、\*、/、%、|、&、^ )。

规则4-6：在双目或者三目运算符（左边和右边）加上空格( <=、>=、==、!=、? :)。

规则4-7：单目运算符如“!”、“~”、“++”、“--”、“\*”（取值）、“&”（取地址）、+（正）、-（负）等前后不加空格。“［］”、“.”、“->”这类操作符前后不加空格。

规则4-8：不要在行尾留有空格。

1. 命名

规则5-1：不使用大小写字母混合写法。

*This\_variable\_is\_a\_temporary\_counter √*

*ThisVariableIsATemporaryCounter  ×*

规则5-2：全局变量需要一个描述性的名字，全局函数也是。

/\*计算活跃用户数量的函数 \*/

*void count\_active\_users( void)*

*{*

*}*

规则5-3：局部变量的名字应当简洁了当。如果在循环中需要一些随机数字，你大可以命名其为 "i" 。只要不会产生歧义，命名为 "loop\_counter" 毫无意义。同样的，"tmp" 可以是任何类型的临时变量。

规则5-4：文件名全部小写。

1. 函数

规则6-1：函数要简短,一个函数只做一件事情。

规则6-2：如果函数相当的复杂，把函数尽量缩短，多造几个辅助函数并给他们几个一看即知的名字，以减少函数复杂性。

规则6-3：函数长度一般不超过2屏，如果函数中的 switch 有很多简单的 case语句, 那么超出2屏也可以。

建议6-4：函数中局部变量不能超过 5~10 个。

规则6-5：函数与函数之间空一行。

规则6-6：在函数原型中，当声明参量时应该将参量的类型和标识符放到一起，目的是增加每一行代码的信息量。

1. 注释

注释是用来说明这段代码是“做什么”，而不是“怎么做”。

规则7-1：注释使用“/\* ... \*/”。单行注释尽量避免使用“//...”。

规则7-2：多于一行（多行）的注释应当遵从以下格式：

规则7-3：该格式对于注释标识符（常量，变量，函数等）同样适用。

规则7-4：不要再一行里面同时声明很多个标识符（无论是用逗号还是分号隔开都是不推荐的），一行一个就可以了。这样你就可以在每一行对每一个标识符进行解释。

规则7-5：中文注释编码格式统一为ANSI