

养成良好的Linux编码风格Linux认证考试 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/619/2021_2022__E5_85_BB_E6_88_90_E8_89_AF_E5_c103_619611.htm

Linux操作系统是一个开源的操作系统，为此你在Linux系统上开发的一个工具软件，包括源代码，可能其他系统管理员也需要用到。为此在编写代码的时候，就需要遵守一定的规则。这不仅是为了方便他人的阅读，也是为了以后自己的维护与升级考虑。具体的来说，笔者认为Linux系统管理员要养成下面的一些好的编码风格。

一、合理防治函数开头的左花括号。根据大部分系统管理员认可的编码风格，往往将函数开头的左括号放到代码页的最左边。要避免将其他的括号(包括左花括号、左括号或者左方括号)放到最左边。这主要是为了便于阅读。因为函数的主体内容往往是有一对花括号括起来的。如果在代码页的最左边只有代表函数的花括号，那么就可以一目了然的看到函数的主体。为此这是提高代码阅读性的一个很好的手段。需要注意的是，这可能跟其他语言的编程风格有所差异。如在Java语言或者C语言平台上，往往将函数主体开头的花括号放在函数的后面。如在main函数后面会直接使用{这个左花括号。不过这不利于程序的阅读，不利于Linux系统管理员找到函数的主体代码。为此如果有其他编程语言使用经验的系统管理员，最好能够改变这种书写习惯。笔者建议，系统管理员还是要将这个花括号放在最左边，并保证在整个代码中，最左边出现的花括号都是代码函数主体的花括号。

二、每个函数开头最好都有一个简短的代表功能的说明。在Linux的功能代码中，其各个功能也都是一个个函数或者程序构成的

。也就是说，在一个代码文件中，可能会有很多个函数构成。那么这些函数主要用来实现什么功能呢？如果不做任何说明的话，那么只有看完函数的全部代码之后才能够了解这个信息。这对于他人阅读源代码会造成比较大的障碍。而且，时间久了之后，可能连系统管理员自己都不知道这个函数时用来实现什么功能的。这对于其后续维护与升级显然是不利的。为此笔者建议各位系统管理员，无论是为自己还是为他人，最好在每个函数或者程序的开头都写上一小段注释。好记性不如烂笔头，这对于提高代码的易读性。另外需要说明的是，由于Linux系统可能对中文的支持并不是很好，为此在写这个注释的时候，最好采用英文书写。因为在一些对中文支持并不是很好的系统中，这个中文会显示为乱码，此时就起不到应有的作用了。如果不懂英linux认证，加入收藏！文的话，那么可能只有使用拼音了。当然这只是笑话。一般Linux系统工程师对于英文需要一定的了解。因为Linux操作系统中的帮助文档都是英文写的。所以这个英文的语言关也是系统管理员必须要解决的关口。另外在对函数进行说明时，最好还需要著名这个函数需要用户传入什么参数，会返回什么样的结果。以及参数、结果的个数等等。这对于代码的编译与维护非常有帮助。而且项目团队中的其他成员如果要引用你这个函数的话，那么不需要查看函数的具体代码。而只需要查看一些这个注释，就可以知道需要传递进去哪些参数。这也是提高项目合作效率的一个手段。

三、If语句使用要规范

。在Linux系统中编写代码时，IF语句是使用的最多的结构之一。这个if语句主要用来实现一些逻辑的判断。虽然这个语句本身比较简单，但是在使用这个语句时，最好也能够遵守一

些规则。虽然这些规则主要是从易读性的角度去考虑的，但是对于编写一个准确的IF结构语句也有所帮助。如根据Linux系统的编程习惯，最好不要在IF的条件中进行赋值。IF语句需要根据某个条件来进行判断该采取什么样的操作。在这个条件中，根据语法是可以在这个条件中对变量进行赋值的。但是这不符合Linux系统下的编程风格。笔者建议，最好在IF结构外部对变量进行赋值，然后再在条件中直接使用这个变量。这更易于控制IF结构。另外，如果在IF语句中使用嵌套的话，可以使用花括号将嵌套的IF ELSE语句括起来，以利于发现这个嵌套语句。在其他应用程序编写过程中，是使用缩进的方式来凸现IF嵌套结构的。但是Linux系统管理员更加喜欢使用花括号。虽然这两个没有实质上的区别，但是笔者还是建议采用花括号。因为这是一个Linux操作系统业内普遍认可的一个编码风格。

四、大小写书写要规范。

虽然函数名或者变量名采用大写字符或者小写字符都是允许的。但是在定义这些名字的时候，大小写最好还是要符合一些常见的规则。如对于函数的名字最好使用小写的英文字符。如果有多个单词构成的话，最好使用下划线来进行分割。而不是采用第一个英文单词大写的方式。大小写混用在某些环境中或许是很不错的解决方式，但是在Linux系统中不行。而对于变量来说，要根据变量的类型来确定使用小写字符还是采用大写字符。一般情况下，如果是系统管理员自己定义的变量，那么最好使用小写字符。如果采用的是系统变量的话，那么可以使用大写的英文字母。如此的话，一看到变量的大写还是小写的状态，就可以判断使用的是用户自定义变量还是系统变量。另外在Linux系统中也需要用到一些常量。根据大家的编

程习惯，往往利用大写字符来定义常量。如系统管理员在学习其他高手编写的代码过程中，可以发现在枚举或者宏中都是利用大写字符来代表常量的。在确定使用大写还是使用小写字符之前，需要明确的一点就是在Linux操作系统中大小写是敏感的。也就是说，Name与name是代表两个不同的变量。为此在系统中规范大小写的格式就显得尤其的重要。由于大小写敏感，所以在同一个名字中混用大小写会让变量或者函数的应用变得很复杂。故如果一个名字有好几个单词构成，那么系统管理员要习惯利用下划线来进行单词的分割，而不是采用第一个单词大写的形式。

五、代码该分行是就需要分行。

虽然Linux操作系统的编译器对于分行没有严格的要求。如几个变量都可以在同一行中定义，或者在同一个行中对多个变量进行赋值等等。只要采用了合适的分割符，就可以编译通过并得到正确的执行。但是如果系统管理员真的这么说的话，那么笔者告诉你，你会被同行笑话的。因为这使你不专业的表现。根据Linux操作系统通用的编程风格，编写代码时该分行是就需要分行。如在变量或者常量定义时，一般都是一行定义个变量或者常量。在编写一些复杂结构语句时，如IF或者循环结构，也往往是采取分行的格式。即使代码很简单，一行就可以写完。但是仍然会根据其不同的部分，将其分割成两行甚至多行。这有利于系统管理员阅读代码。另在程序的不同部分，也最好能够采取分行的方式来进行隔离。如函数与函数之间，变量声明与函数之间，注释与代码之间等等。可以加入一到两行的空行，来提高阅读的便利性。反正这不会对程序的性能造成影响。类似的编码风格还有很多。如两个不同运算级别的操作符号组好不要在同行中出现。

不要跨行声明多个变量，即变量的声明最好放在同一个地方，以方便同一管理，等等。这些规则需要系统管理员他人编写的代码中去体会与总结。多看看一些专家级人物写的代码，往往可以给我们带来很大的收获。最后要说的是，这些规则并不是强制性的。也就是说你即使不遵守这些规则的话，Linux系统编译器也不会认为你的代码有问题，可以正确执行。但是每个行业除了要遵守一些强制的法律法规之外，还需要遵守一些行业性规范。而这些编码的风格就是Linux操作系统的行业性规范。如果系统管理员不遵守的话，可能会被大家排斥。所以，系统管理员最好还是花点时间和精力去学习这些行业规范。100Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com