[**C/C++代码格式优化工具----astyle**](http://www.cnblogs.com/zhaoshixin/archive/2011/12/02/2272076.html)

常用的C/C++代码格式优化工具有两个，一是老牌的indent，再一个就是astyle了。

astyle不但可以对C/C++进行格式优化，还可以处理Java和C#。版本一直在保持更新，很不容易，截止2011-12-02，已经升级到V2.02.1。  
通过命令““astyle -V”可以查询版本信息，最新版本的源代码和可执行文件可以在此处【[下载](http://sourceforge.net/projects/astyle/)】。

网络上已经有很多博客介绍过astyle，如《[在VS 2008中使用astyle代码格式化工具](http://www.cnblogs.com/deagle/articles/1968895.html)》。  
不过介绍中对astyle的参数存在些许偏差，可能是版本升级后参数发生了变化。  
通过命令“astyle --help”可以获取所有参数的介绍（帮助信息里还有错字呢，有兴趣大家可以找找看）。

下面的命令可以一次性格式化某个目录下所有的源文件和头文件，非常好用，标记一下。  
**for /R %f in (\*.cpp;\*.c;\*.h) do astyle --style=ansi  "%f"**

基于V2.02版本，astyle主要支持的参数有：

**Style-格式配置：**  
最常用的就是ansi或或kr格式，实际上，kr，stroustrup和linux这三种格式是非常接近的了，试了好几个文件，只有非常微小的区别，可以忽略不计。

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| stype 选项 | --style=allman --style=ansi --style=bsd --style=break -A1 | --style=java --style=attach -A2 | --style=kr --style=k&r --style=k/r -A3 | --style=stroustrup -A4 | --style=whitesmith -A5 | --style=banner -A6 |
| 代码风格 | int Foo() {     if (isBar)     {         bar();         return 1;     }     else     {         return 0;     } } | int Foo() {     if (isBar) {         bar();         return 1;     } else {         return 0;     } } | int Foo() {     if (isBar) {         bar();         return 1;     } else {         return 0;     } } | int Foo() {     if (isBar) {         bar();         return 1;     } else {         return 0;     } } | int Foo()     {     if (isBar)         {         bar();         return 1;         }     else         {         return 0;         }     } | int Foo() {     if (isBar) {         bar();         return 1;         }     else {         return 0;         }     } |
| stype 选项 | --style=gnu -A7 | --style=linux -A8 | --style=horstmann -A9 | --style=1tbs -A10 | --style=pico -A11 | --style=lisp -A12 |
| 代码风格 | int Foo() {     if (isBar)         {             bar();             return 1;         }     else         {             return 0;         } } | int Foo() {     if (isBar) {         bar();         return 1;     } else {         return 0;     } } | int Foo() {   if (isBar)     {   bar();         return 1;     }     else     {   return 0;     } } | int Foo() {     if (isBar) {         bar();         return 1;     } else {         return 0;     } } | int Foo() {   if (isBar)     {   bar();         return 1; }     else         return 0; } | int Foo() {     if (isBar) {         bar();         return 1; }     else         return 0; } |

其他常用的参数：

|  |  |
| --- | --- |
| -C | 类中public,pretected,private关键字，一个tab的缩进 |
| -S | switch中case关键字，一个tab的缩进 |
| -K | switch中case关键字，无缩进 |
| -N | 被namespace包含的block，一个tab的缩进 |
| -w | 格式化多行的宏定义 |
| -c | 将tab转化为对应个数的空格 |
| --mode=c | 格式化的是C/C++的源文件或者头文件（缺省值） |
| --mode=java | 格式化的是JAVA的源文件 |
| --suffix=#### | 将原始文件保存为“####”后缀，而不是“orig” |
| --suffix=none | 不保存原始文件 |
| --exclude=#### | 优化时不包含“####”文件或目录 |
| -Z | 修改后保持文件的修改时间不变 |
| -X | 将错误信息输出到标准输出设备（stdout），而不是标准错误设备（stderr） |
| -Q | 只显示格式化前后发生变化的文件 |
| -q | 不输出任何信息 |
| -z1 | 使用windows版本的回车符(CRLF) |
| -z2 | 使用linux版本的回车符(LF) |
| --help | 显示帮助信息 |
| -v | 显示版本信息 |

分类: [软件技巧](http://www.cnblogs.com/zhaoshixin/category/339826.html)