

# C语言中的\_\_FILE\_\_、\_\_LINE\_\_和#line（用于打印当前函数及行数）

转载 IT之路 2018-04-02 08:54:47 1249 收藏 1

分类专栏： [c/c++](#) 文章标签： [\\_\\_LINE\\_\\_](#) [\\_\\_FILE\\_\\_](#)

C语言中的\_\_FILE\_\_用以指示本行语句所在源文件的文件名，举例如下（test.c）：

1. `#include <stdio.h>`
2. `int main()`
3. `{`
4. `printf("%s\n",__FILE__);`
5. `}`

在gcc编译生成a.out，执行后输出结果为：

`test.c`

在windows的vc6.0下编译执行结果为：

`c:\documents and settings\administrator\桌面\test.c`

C语言中的\_\_LINE\_\_用以指示本行语句在源文件中的位置信息，举例如下：

1. `#include <stdio.h>`
- 2.
- 3.
- 4.
5. `main()`
6. `{`
7. `printf("%d\n",__LINE__);`
8. `printf("%d\n",__LINE__);`
9. `printf("%d\n",__LINE__);`
10. `};`

该程序在linux用gcc编译，在windows的vc6.0下编译都可以通过，执行结果都为：

`7`  
`8`  
`9`

还可以通过语句#line来重新设定\_\_LINE\_\_的值，举例如下：

```
1. #include <stdio.h>

2.

3.

4. #line 200 //指定下一行的__LINE__为200

5. main()

6. {

7. printf("%d\n",__LINE__);

8. printf("%d\n",__LINE__);

9. printf("%d\n",__LINE__);

10. };
```

编译执行后输出结果为：

```
202
203
204
```

---

---

另外gcc还支持\_\_func\_\_,它指示所在的函数，但是这个关键字不被windows下的vc6.0支持，举例如下：

```
1. #include <stdio.h>

2. void main()

3. {

4. printf("this is print by function %s\n",__func__);

5. }
```

其编译后输出结果为

```
this is print by function main
```

注意“#line”、“\_\_LINE\_\_”、“\_\_FILE\_\_”及“\_\_func\_\_”都是大小写敏感的。