



微信扫一扫  
关注该公众号

收录于合集

#C++小知识 16 #C++入门到进阶 13

## 宏定义污染与规避

### 问题

众所周知，宏定义是没有 名称空间 的限制的，所以宏定义很容易导致的命名冲突。比如下代码：

```
// msvc 2022
#include <iostream>
#include <windows.h>

int main()
{
    constexpr int m = std::min(1, 0);
    return 0;
}
```

VC「编译输出」

```
error C2589 : “(” : “::”右边的非法标记
error C2062 : 意外的类型“unknown - type”
error C2059 : 语法错误:“)”
```

你能看出来错误在哪吗？

没错，就是 windows.h 头文件里面有一个宏定义：

```
#ifndef min
#define min(a,b) ((a) < (b)) ? (a) : (b))
#endif
```

我们上面的代码被展开成如下形式：

```
constexpr int m = std::(((1) < (0)) ? (1) : (0))
```

### 规避

可以通过 加括号、去定义、定义空宏 等方式避免污染。

#### 加小括号

我们既然不能改源码，自然要从我们自己的代码着手。我们将冲突的代码加上 ( ) 如下：

```
constexpr int m = (std::min)(1, 0);
```

编译器看到 (std::min) 就不使用宏 min 替换了。

#### 去定义

小心项目中其他有用到的情况。

```
#ifdef min
#undef min
#endif
```

#### 换名字

鉴于我们使用的都是三方库，这个办法无效。

### 趣谈

考虑如下定义

```
#define private public
```

收录于合集 #C++小知识 16

上一篇

C++中switch...case绕过变量初始化

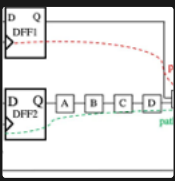
下一篇

is\_same 判断类型是否相同

阅读原文

喜欢此内容的人还喜欢

STA介绍——setup和hold互卡问题  
读芯树



【独立站社群聚集地】全球最受欢迎的聊天应用程序  
跨境x小霸x王



金蝶云·苍穹V5.0新特性精选文章合集 之 集成服务云篇  
PaaS平台那些事

