

C 中extern “ C ” 含义深层探索 PDF转换可能丢失图片或格式  
， 建议阅读原文

[https://www.100test.com/kao\\_ti2020/133/2021\\_2022\\_C\\_\\_\\_E4\\_B8\\_](https://www.100test.com/kao_ti2020/133/2021_2022_C___E4_B8_ADex-te_c97_133800.htm)

[ADex-te\\_c97\\_133800.htm](https://www.100test.com/kao_ti2020/133/2021_2022_C___E4_B8_ADex-te_c97_133800.htm) 1.引言 C 语言的创建初衷是 “ a better C ” ，但是这并不意味着C 中类似C语言的全局变量和函数所采用的编译和连接方式与C语言完全相同。作为一种欲与C兼容的语言，C 保留了一部分过程式语言的特点（被世人称为 “ 不彻底地面向对象 ” ），因而它可以定义不属于任何类的全局变量和函数。但是，C 毕竟是一种面向对象的程序设计语言，为了支持函数的重载，C 对全局函数的处理方式与C有明显的不同。 2.从标准头文件说起 某企业 曾经给出如下的一道面试题：面试题 为什么标准头文件都有类似以下的结构？

```
#ifndef __INCvxWorksh #define __INCvxWorksh #ifdef  
__cplusplus extern "C" { #endif /* ...*/ #ifdef __cplusplus } #endif  
#endif /* __INCvxWorksh */
```

分析显然，头文件中的编译宏 “ #ifndef \_\_INCvxWorksh、 #define \_\_INCvxWorksh、 #endif ” 的作用是防止该头文件被重复引用。那么 #ifdef \_\_cplusplus extern "C" { #endif #ifdef \_\_cplusplus } #endif 的作用又是什么呢？我们将在下文一一道来。 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](http://www.100test.com)