

教你理解复杂的C/C 声明 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/134/2021_2022__E6_95_99_E4_BD_A0_E7_90_86_E8_c97_134311.htm 原文：

http://www.codeproject.com/cpp/complex_declarations.asp 作者：Vikram A Punathambekar 介绍 曾经碰到过让你迷惑不解、类似于 `int * (* (*fp1) (int)) [10]`. 这样的变量声明吗？本文将由易到难，一步一步教会你如何理解这种复杂的C/C 声明：我们将从每天都能碰到的较简单的声明入手，然后逐步加入 `const` 修饰符和 `typedef`，还有函数指针，最后介绍一个能够让你准确地理解任何C/C 声明的“右左法则”。需要强调一下的是，复杂的C/C 声明并不是好的编程风格；我这里仅仅是教你如何理解这些声明。注意：为了保证能够在同一行上显示代码和相关注释，本文最好在至少1024x768分辨率的显示器上阅读。来源：www.examda.com 基础 让我们从一个非常简单的例子开始，如下：`int n`. 这个应该被理解为“declare n as an int”（n是一个int型的变量）。接下去来看一下指针变量，如下：`int *p`. 这个应该被理解为“declare p as an int *”（p是一个int *型的变量），或者说p是一个指向一个int型变量的指针。我想在这里展开讨论一下：我觉得在声明一个指针（或引用）类型的变量时，最好将*（或amp.，如下：`int **p1`. // p1 is a pointer to a pointer to an int. `int *amp.*p3`. // ERROR: Pointer to a reference is illegal. `int amp.p4`. // ERROR: Reference to a reference is illegal. 注：p1是一个int类型的指针的指针；p2是一个int类型的指针的引用；p3是一个int类型引用的指针（不合法！）；p4是一个int类型引用的引用（不合法

！)。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