

typedef函数指针的用法（附例子）计算机等级考试 PDF转换可能丢失图片或格式，建议阅读原文

[https://www.100test.com/kao\\_ti2020/645/2021\\_2022\\_typedef\\_E5\\_87\\_BD\\_c97\\_645690.htm](https://www.100test.com/kao_ti2020/645/2021_2022_typedef_E5_87_BD_c97_645690.htm) 在网上搜索函数指针，看到一个例子。开始没看懂，想放弃，可是转念一想，这个用法迟早要弄懂的，现在多花点时间看懂它，好过以后碰到了要再花一倍时间来弄懂它。其实很多时候都是这样，如果每次到难一点的内容，总想着下次我再来解决它，那就永远也学不到东西。后面那个例子加了注释，是我对这种用法的理解，希望对新手有所帮助。进入正文：代码简化, 促进跨平台开发的目的. typedef 行为有点像 #define 宏，用其实际类型替代同义字。不同点：typedef 在编译时被解释，因此让编译器来应付超越预处理器能力的文本替换。用法一：typedef int (\*MYFUN)(int, int). 这种用法一般用在给函数定义别名的时候 上面的例子定义MYFUN 是一个函数指针, 函数类型是带两个int 参数, 返回一个int 在分析这种形式的定义的时候可以用下面的方法: 先去掉typedef 和别名, 剩下的就是原变量的类型. 去掉typedef 和MYFUN以后就剩: int (\*)(int, int) 用法二：typedef给变量类型定义一个别名. typedef struct{ int a. int b. }MY\_TYPE. 这里把一个未命名结构直接取了一个叫MY\_TYPE的别名, 这样如果你想定义结构的实例的时候就可以这样: MY\_TYPE tmp. 第二种用法：typedef 原变量类型 别名 简单的函数指针的用法 //形式1：返回类型 (\*函数名) (参数表) char (\*pFun)(int). //typedef char(\*pFun)(int) 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)