

c_c 支持可变参数的函数计算机等级考试 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_c_c___E6_94_AF_E6_8C_c97_645886.htm 一、为什么要使用可变参数的函数？

一般我们编程的时候，函数中形式参数的数目通常是确定的，在调用时要依次给出与形式参数对应的所有实际参数。但在某些情况下希望函数的参数个数可以根据需要确定，因此c语言引入可变参数函数。这也是c功能强大的一个方面，其它某些语言，比如fortran就没有这个功能。典型的可变参数函数的例子有大家熟悉的printf（）、scanf（）等。

二、c/c 如何实现可变参数的函数？为了支持可变参数函数，C语言引入新的调用协议，即C语言调用约定 __cdecl。采用C/C语言编程的时候，默认使用这个调用约定。如果要采用其它调用约定，必须添加其它关键字声明，例如WIN32 API使用PASCAL调用约定，函数名字之前必须加__stdcall关键字。采用C调用约定时，函数的参数是从右到左入栈，个数可变。由于函数体不能预先知道传进来的参数个数，因此采用本约定时必须由函数调用者负责堆栈清理。举个例子：//C调用约定函数 int __cdecl Add(int a, int b) { return (a+b); } 函数调用：Add(1, 2). //汇编代码是：push 2.参数b入栈 push 1.参数a入栈 call @Add.调用函数。其实还有编译器用于定位函数的表达式这里把它省略了 add esp,8.调用者负责清栈 如果调用函数的时候使用的调用协议和函数原型中声明的不一致，就会导致栈错误，这是另外一个话题，这里不再细说。另外c/c 编译器采用宏的形式支持可变参数函数。这些宏包括va_start、va_arg和va_end等。之所以这么做，是为了增加程序的可移植性。

屏蔽不同的硬件平台造成的差异。支持可变参数函数的所有宏都定义在 `stdarg.h` 和 `varargs.h` 中。例如标准ANSI形式下，这些宏的定义是：
`typedef char * va_list. //字符串指针 #define`
`_INTSIZEOF(n) ((sizeof(n) sizeof(int) - 1)`
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