

计算机二级辅导:c_c 内存划分计算机等级考试 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__E8_AE_A1_E7_AE_97_E6_9C_BA_E4_c97_645867.htm

一、一个经过编译的C/C++的程序占用的内存分成以下几个部分：1、栈区（stack）：由编译器自动分配和释放，存放函数的参数值、局部变量的值等，甚至函数的调用过程都是用栈来完成。其操作方式类似于数据结构中的栈。2、堆区（heap）：一般由程序员手动申请以及释放，若程序员不释放，程序结束时可能由OS回收。注意它与数据结构中的堆是两回事，分配方式类似于链表。3、全局区（静态区）（static）：全局变量和静态变量的存储是放在一块的，初始化的全局变量和静态变量在一块区域，未初始化的全局变量和未初始化的静态变量在相邻的另一块区域。程序结束后由系统释放空间。4、文字常量区：常量字符串就是放在这里的。程序结束后由系统释放空间。5、程序代码区：存放函数体的二进制代码。下面的例子可以完全展示不同的变量所占的内存区域：

```
//main.cpp
int a = 0. //全局初始化区
char *p1. //全局未初始化区
main() {
    int b. //栈中
    char s[] = "abc". //栈中
    char *p2. //栈中
    char *p3 = "123456". //123456\0在常量区，p3在栈上
    static int c = 0; //全局（静态）初始化区
    //以下分配得到的10和20字节的区域就在堆区
    p1 = (char *)malloc(10).
    p2 = new char[20]. //(char *)malloc(20).
    strcpy(p1, "123456"). //123456\0放在常量区，编译器可能会将它与p3所指向的"123456"优化成一个地方。
}
```

二、栈（stack）和堆（heap）具体的区别。1、在申请方式上栈（stack）：现在很多人都称之为堆栈，这个时候实际上还是指

的栈。它由编译器自动管理，无需我们手工控制。例如，声明函数中的一个局部变量 `int b` 系统自动在栈中为 `b` 开辟空间；在调用一个函数时，系统自动的给函数的形参变量在栈中开辟空间。

堆 (heap)：申请和释放由程序员控制，并指明大小。容易产生 `memory leak`。在C中使用 `malloc` 函数。如：`p1 = (char *)malloc(10)`。在C中用 `new` 运算符。如：`p2 = new char[20]`。// `(char *)malloc(10)`。但是注意 `p1` 本身在全局区，而 `p2` 本身是在栈中的，只是它们指向的空间是在堆中。

2、申请后系统的响应上

栈 (stack)：只要栈的剩余空间大于所申请空间，系统将为程序提供内存，否则将报异常提示栈溢出。

堆 (heap)：首先应该知道操作系统有一个记录空闲内存地址的链表，当系统收到程序的申请时，会遍历该链表，寻找第一个空间大于所申请空间的堆结点，然后将该结点从空闲结点链表中删除，并将该结点的空间分配给程序。另外，对于大多数系统，会在这块内存空间中的首地址处记录本次分配的大小，这样，代码中的 `delete` 或 `free` 语句才能正确的释放本内存空间。另外，由于找到的堆结点的大小不一定正好等于申请的大小，系统会自动的将多余的那部分重新放入空闲链表中。

3、申请大小的限制

栈 (stack)：在Windows下，栈是向低地址扩展的数据结构，是一块连续的内存的区域。这句话的意思是栈顶的地址和栈的最大容量是系统预先规定好的，在WINDOWS下，栈的大小是 2M（也有的说是 1M，总之是一个编译时就确定的常数），如果申请的空间超过栈的剩余空间时，将提示 `overflow`。因此，能从栈获得的空间较小。例如，在VC6下面，默认的栈空间大小是 1M（好像是，记不清楚了）。当然，我们可以修改：打开工程，依次操作菜单如

下：Project- 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