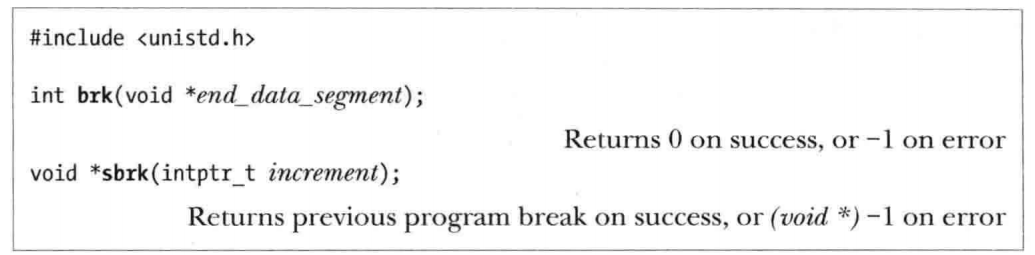
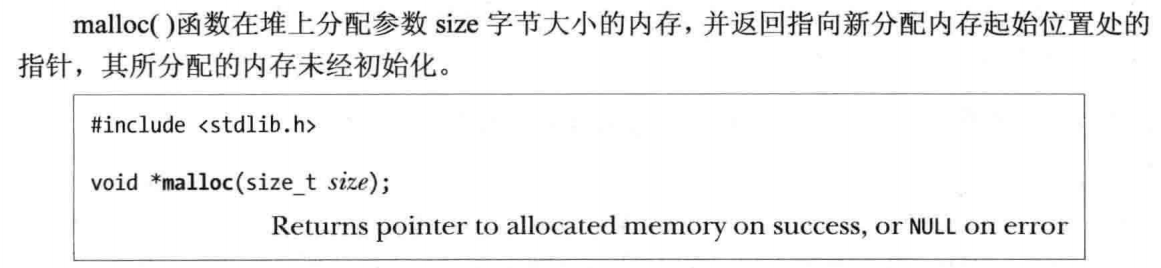
第七章

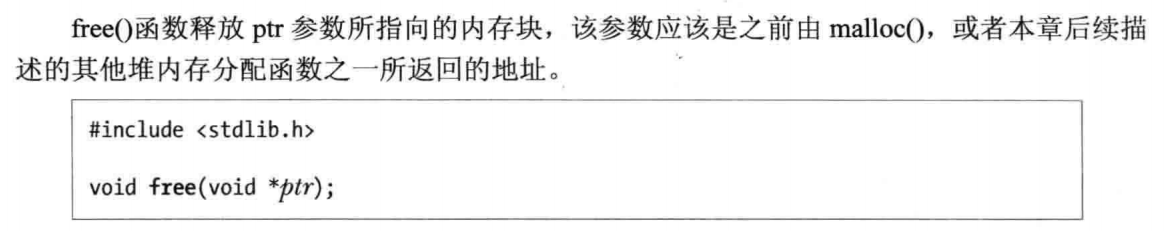
1. 在堆上分配内存：以下函数作用是调整program break（通常将堆的边界称为这个）。



注意，若sbrk调用成功，返回前一个program break的地址，也就是program break增加了，返回的就是这个新分配内存起始位置的指针。

1. C语言标准下堆分配内存：

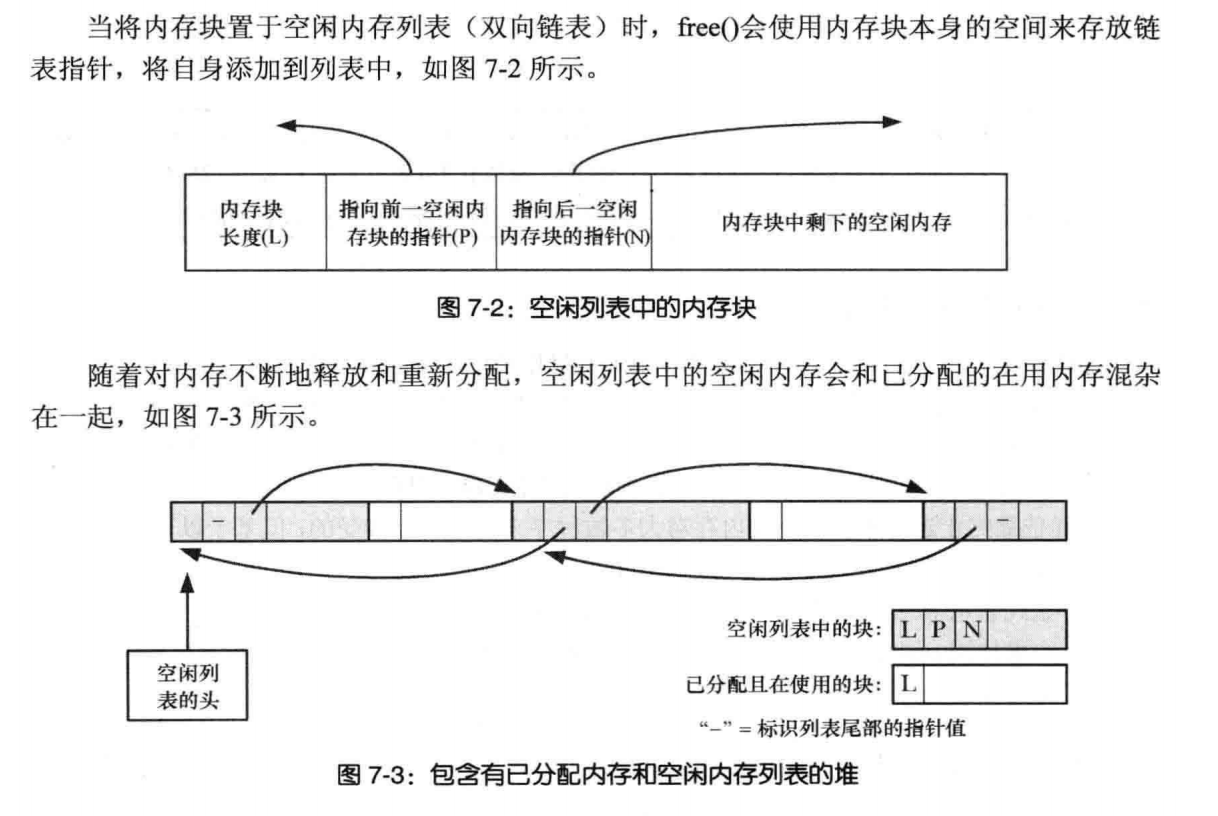




注意，当堆顶空闲内存足够大的时候，free函数会调用sbrk来降低program break的地址。

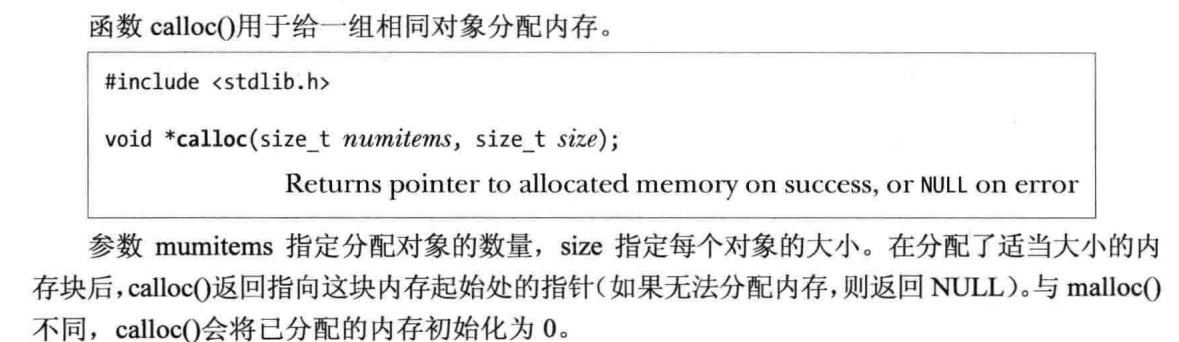
1. malloc函数和free的实现：malloc和free会维护一个内存块列表，malloc实现的时候会扫描free所释放的空闲内存块列表，以某种策略找到合适的内存块。

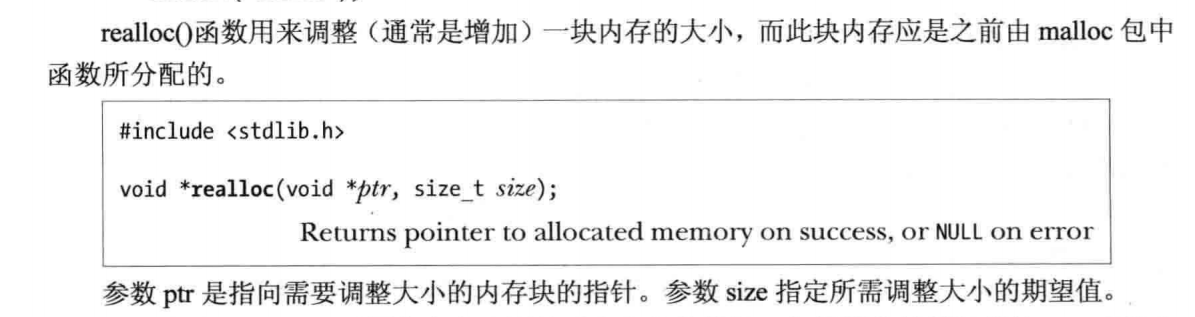




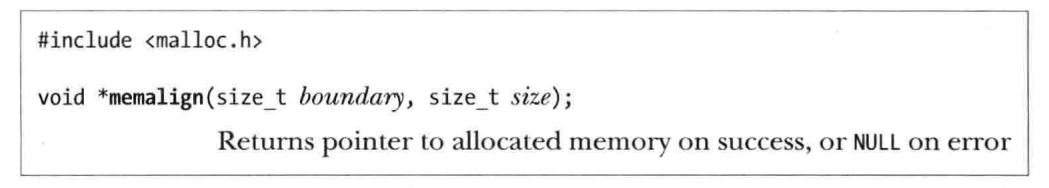
Free函数的实现如上图，实则在维护一个双向列表。

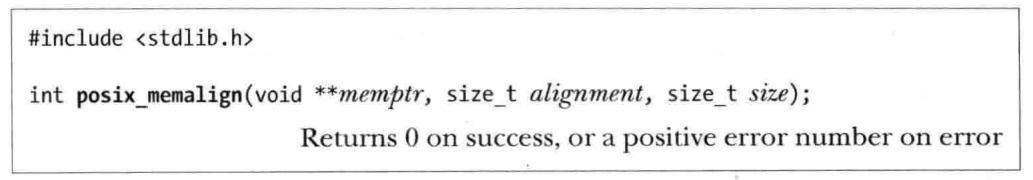
1. 由于malloc使用会导致错误，glibc提供了一些调试库。
2. 在堆上分配内存的其他方法：



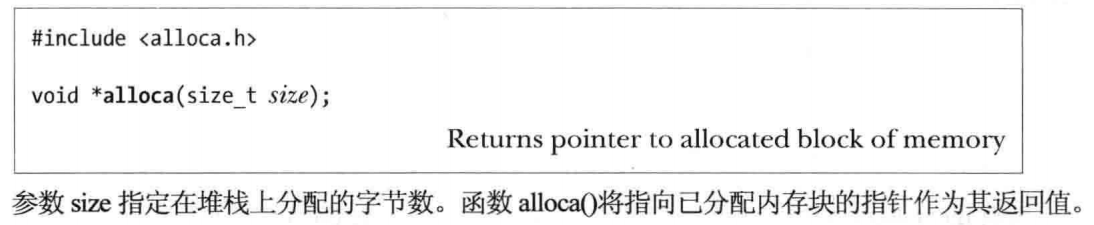


1. 分配对齐内存的函数：





1. 在堆栈上（也就是栈的说法）分配内存



这个函数有一些注意事项：不能也无需用free销毁内存，他会自动销毁（栈的原因），不能在函数参数里使用这个函数，因为他会在函数参数中间申请空间，而函数引用参数是有固定规律的。此外，函数错误不会返回NULL，注意接收某些信号判别错误。