

2020/04/29_第20课_动态内存分配

笔记本: C

创建时间: 2020/4/29 星期三 15:54

作者: ileemi

- [动态内存分配相关函数](#)

动态字符串管理 就是模拟堆的数据存储

堆会做一个分配表, 记录数据的记录起始位置 长度

堆分配表

分配表 -> 磁盘文件

见缝插针: 高效利用了散碎的空闲空间

簇 链

堆和栈都支持动态增长

栈的增长是线性的, 生长放行向0生长

堆是哪里有空间就用哪里

Dubug版本下

已分配空间初值会置为CDCD 屯

未分配空间会置为0xFEEE葇 或者0xDDDD铅

realloc

- 1、判断当前内存是否可以存放
- 2、重新开辟空间, 将原来的数据拷贝到新开辟的空间
- 3、将源数据区进行释放

windows系统中堆用到的函数

HeapCreate 打造堆环境, 建造大楼, 返回值为堆开始的地址

HeapAlloc 租用房间

HeapFree 退租

HeapDestroy 拆毁大楼

动态内存分配相关函数

malloc() 申请堆空间, 用于返回指向数组的指针, 指向结构的指针等。所以通常该函数的返回值会被强制转换为匹配的类型。

函数原型: void *malloc(unsigned int num_bytes); //分配长度为num_bytes字节的内存块

calloc() 在内存的动态存储区中分配n个长度为size的连续空间, 函数返回一个指向分配起始地址的指针; 如果分配不成功, 返回NULL。

函数原型: void *calloc(size_t n, size_t size);

realloc() 重新申请空间大小 改变原来sizeof的空间大小 (修改)

函数原型: void *realloc(void memblock, size_t size);

指针名 = (数据类型) realloc (要改变内存大小的指针名, 新的大小)。

free() free()函数的参数是之前malloc()函数返回的地址, 该函数释放之前malloc分配的内存(或calloc、realloc)函数给指针变量分配的内存空间。

_malloc_dbg 函数

代码区: 存放程序代码部分

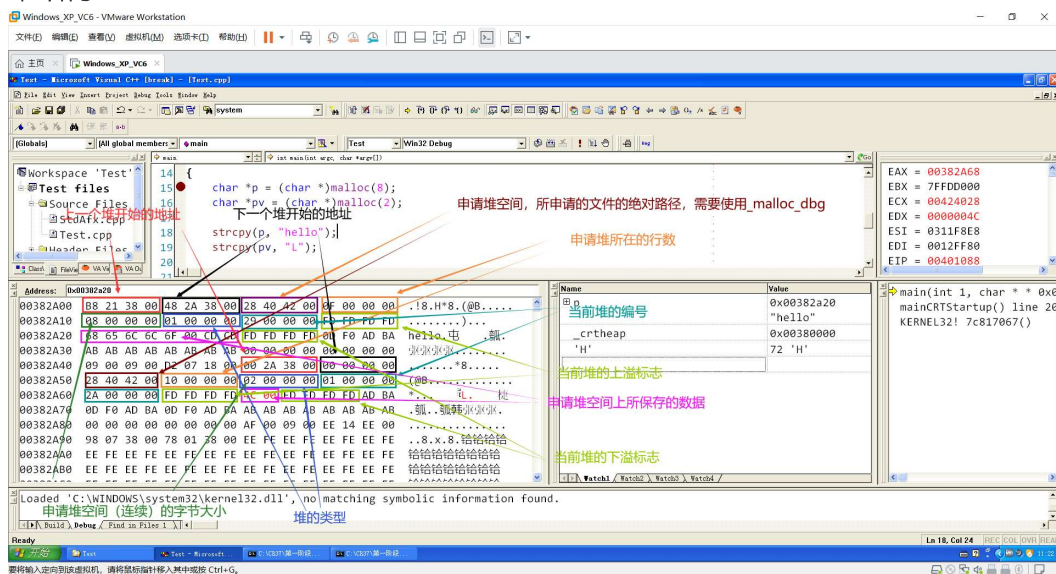
数据区: 存放程序的全局数据和静态数据

堆区: 存放程序动态申请的数据

栈区: 存放程序的局数据和参数

一块堆空间是存在附加数据的

堆结构:



VC++ 6.0 外的高版本 IDE 在内存中, 申请堆的字节大小和堆的类型会交换位置