

内存结构

```
code
|-- re
data      -|\
|-- rw    -|- Init'd and Uninit - global and static
|-- r      -|/
stack
|
heap
```

- 数据段(**data**)
 - 通常是指用来存放程序中**已初始化的**全局变量的一块内存区域，属于静态内存分配
 - 一般来说，拥有**只读**或者**读写**属性
- 代码段(**code**)
 - 通常代码段和只读数据段合成为文本段(**text segment**)，包含实际要执行的代码（机器指令）和常量
 - 一般来说，有用**可读可执行**属性

调试

在IDE中调试时，首先可以拉出以下等窗口




- 监视窗口
- 内存窗口
- 堆栈调用窗口
- 寄存器窗口


在vs2017中调试时，有以下快捷键可以帮助我们

- **F11**逐语句，可以跟进到函数内部执行
- **F10**逐过程，可以跳过不关心的函数，每次执行一条语句
- **Shift+F11**跳出，可以跳出当前的函数，返回调用层

当我们需要观察某变量的变化时，可以在监视窗口输入该变量的名字，如果对某块内存感兴趣时，可以在内存窗口输入地址。

在堆栈调用窗口中，我们可以看到函数调用情况

名称	值	类型
 lo4_ch	-858993460	int
 hi4_ch	-858993460	int
 *ch	67 'C'	char

名称	语言
 encryption.exe!encrypt(char * ch) 行 26	C
encryption.exe!main(int argc, char ** argv) 行 17	C
encryption.exe!invoke_main() 行 78	C++
encryption.exe!_sclr_common_main_seh() 行 288	C++
encryption.exe!_sclr_common_main() 行 331	C++
encryption.exe!mainCRTStartup() 行 17	C++
kernel32.dll!@BaseThreadInitThunk@12()	未...
ntdll.dll!_RtlUserThreadStart()	未...
ntdll.dll!_RtlUserThreadStart@8()	未...

地址: 0x01095AB5		⌵ Ⓜ Ⓜ 自动
0x01095AB5	43 52 33 33 44 75 73 69 68 61 6e 67 00 fd fd fd 61 00 92 70 98 1d 4b ff 00 00 a8 4c 09 01 c0 00 09 01 06 70 91 08 59 ff 00 08 4c 0c d8 77 a0 59 09 01 54	CR33Dusihang.????a.?p?.K...?L...?p?.Y...L.?w?Y...T
0x01095AE9	0c d0 77 a8 59 09 01 b0 59 09 01 c8 4c 09 01 00 00 4a 0f b0 b1 50 0f 00 40 17 00 42 00 44 00 90 5e 09 01 1a 00 1c 00 b0 5e 09 01 ec a2 08 00 06 00 00 e8	.?w?Y...?Y...?L...?P...@...B.D.?^...?^...?P...?P...?
0x01095B1D	0a d0 77 e8 0a d8 77 1b e4 1f 73 00 00 00 00 00 00 90 5b 09 01 90 5b 09 01 00 00 00 00 00 17 00 44 11 be 77 80 49 09 01 08 5a 09 01 20	.?w?.?w...?s.....?[...??.?.....D.?wEI...Z...
0x01095B51	4d 09 01 00 00 00 00 00 00 00 0d 49 09 01 00 00 4a 0f 00 00 00 90 27 02 15 b3 c9 d4 01 b9 30 00 2b 00 00 00 00 00 00 02 00 00 00 00 00 00	M.....?I.....??.?P??..?P?+.....
0x01095B85	00 00 00 97 70 91 19 4d ff 00 0c 34 5b 09 01 34 5b 09 01 00 00 00 02 00 00 00 00 00 03 00 00 00 00 00 d0 5b 09 01 09 00 00 00 00 00 00 03	...?p?.M...4[...4[.....?.....?.....
0x01095BB9	00 00 00 00 00 00 93 70 91 1d 5c ff 00 08 c8 5b 09 01 90 5b 09 01 98 4c 09 01 58 4c 09 01 9d 70 91 13 58 ff 00 08 34 fa fb 00 03 00 00 00 00 00 00 06?p?...?....?L...XL...?p?.X...4??.....