操作系统课程设计实验报告

**实验四 内存监视器模拟实现**

指导教师：陆慧梅老师

班 级：07111507

学 号：1120151880

姓 名：廖汉龙

邮 箱：[liaohanlong@outlook.com](mailto:liaohanlong@outlook.com)

2018 年 3 月 27 日

**一、实验目的**

掌握在Windows系统中设计内存监视器

(1). 熟悉Windows环境下查看内存信息的系统函数

学习Windows系统下查看内存信息的有关系统函数，掌握函数作用，理解函数参数代表的具体内容。

(2). 掌握查看内存状态的方法。

学会查看系统内存大小，内存占用率，系统地址空间的布局等信息，以及某个进程的虚拟地址空间使用情况和工作集信息。

**二、实验内容**

Windows设计一个内存监视器，要求：实时地显示当前系统中内存的使用情况，包括系统地址空间的布局，物理内存的使用情况；实时显示实验二进程控制(ParentProcess.exe)的虚拟地址空间布局和工作集信息 。

相关的系统调用：GetSystemInfo, VirtualQueryEx,VirtualAlloc, GetPerformanceInfo, GlobalMemoryStatusEx ...

三、实验环境及配置方法

操作系统：Windows 10 家庭中文版

编译环境：Visual Studio 2015

gcc 4.9.2

**四、实验步骤与内容**

**1.本实验基本API介绍**

* **GetSystemInfo() 获得当前系统的一些特征信息**

|  |
| --- |
|  |

其中，SYSTEM\_INFO 结构定义如下：

|  |
| --- |
|  |

该字段中，有四个字段与内存有关

dwPageSize 为内存页的大小，当计算机CPU为x86时，该值就为4096

lpMinimumApplicationAddress 为每个进程可用地址空间的最小内存地址

lpMaximumApplicationAddress 为每个进程可用的私有地址空间最大的内存地址

dwAllocationGranularity 为能够保留地址空间区域的最小单位，win32默认为64kb

* **GlobalMemoryStatus() 检索当前系统使用的物理与虚拟内存的信息**

|  |
| --- |
|  |

其中MEMORYSTATUS结构体的定义如下：

|  |
| --- |
|  |

此接口不返回一个值

* **CreateToolhelp32Snapshot() 获取指定进程的快照，以及这些进程使用的堆，模块和线程**

|  |
| --- |
|  |

如果函数成功，它将返回一个打开的句柄到指定的快照。

如果该函数失败，则返回INVALID\_HANDLE\_VALUE。要获得扩展的错误信息，请调用 GetLastError。可能的错误代码包括ERROR\_BAD\_LENGTH。这个接口和其他几个接口一起使用，并且，这里边还会有一个结构PROCESSENTRY32

使用PROCESSENTRY32来存储进程的全部信息，其定义如下：

|  |
| --- |
|  |

在使用了CreateToolhelp32Snapshot()获得了快照的句柄之后，使用Process32First(), Process32Next()等结构取得快照中的第一个，下一个进程的信息，赋值给当前定义的PROCESSENTRY32结构体

* **OpenProcess() 打开一个现有的本地进程对象**

|  |
| --- |
|  |

如果函数成功，返回值是指定进程的打开句柄。

如果函数失败，返回值为NULL。要获得扩展的错误信息，调用 GetLastError。

* **ZeroMemory() 用0填充一块内存**

|  |
| --- |
|  |

这个宏没有返回值

* **VirtualQueryEx() 检查进程虚拟内存的当前信息**

|  |
| --- |
|  |

如果调用函数成功，则返回写入结构lpBuffer的字节数

需要说明一下进程虚拟内存空间的基本信息结构 MEMORY\_BASIC\_INFORMATION

|  |
| --- |
|  |

* **GetMuduleFilenName() 检验可执行的映像**

|  |
| --- |
|  |

如果函数成功，则返回值是复制到缓冲区的字符串的长度（以字符为单位），不包括终止空字符。如果缓冲区太小，无法容纳模块名，字符串被截断为n大小字符，包括终止空字符，该函数返回n大小和功能设置的最后一个错误ERROR\_INSUFFICIENT\_BUFFER。

**2.实验步骤**

**1.实验的思路图**

因为这个实验的主要需要实现的功能是调用Windows 系统中的接口，所以操作过程较为简单，下图是程序中需要实现的功能和每个部分使用的主要的接口函数的说明。

|  |
| --- |
|  |

**2.分块说明**

（1）主函数的主要部分，为了实现如图所示的选择功能，需要在主函数中进行简单的条件语句的设计

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

（2）.功能1 查看内存配置信息

这部分代码主要是要实现查看内存的信息，定义了函数GetSystemInfo\_List(), 在函数中使用的API为GetSystemInfo(),核心调用的代码如下：

|  |
| --- |
|  |

再将si中的各项打印。

（3） 功能2 查看内存使用信息

这部分代码中亚哟要实现查看当存的使用信息，定义了函数GetMemory\_List()， 在函数中使用的API为GlobalMemoryStatu(),核心调用代码如下：

|  |
| --- |
|  |

将stat 中的各项打印。

（4）功能3 查看当前运行进程的信息

功能4 首先打印出当前快照中的所有进程的基本信息，用户可以输入序号或者进程的PID查询选中的进程的按照存储块给出的详细信息。

|  |
| --- |
|  |

功能3的实现过程的流程如下图所示，主要针对每一个进程打印三部分信息，块的状态，如已经提交或者空闲状态；块的权限，可读，可写等；以及邻近页面物理存储器类型指的是与给定地址所在页面相同的存储器类型，如Image，Private以及 Mapped类型等。

|  |
| --- |
|  |

* 打印块的状态主要代码如下：

|  |
| --- |
|  |

* 打印块权限自定义了函数ShowProtection()，定义如下：

|  |
| --- |
|  |

* 打印存储器类型主要代码如下：

|  |
| --- |
|  |

**五、实验结果分析**

**（1）当输入1 时，查看内存配置信息**

|  |
| --- |
|  |

**（2）当输入2 时，查看内存使用信息**

|  |
| --- |
|  |

**（3）当输入3时，查看当前运行进程信息**

|  |
| --- |
|  |

在3下有两种方式，可以查看指定进程的详细信息，输入1和输入2

当输入1时的结果如下图所示

|  |
| --- |
|  |

|  |
| --- |
|  |

当输入2 时，结果为

|  |
| --- |
|  |

**（4）程序提供容错机制**

当前的指令有误或者当前需要查询的进程不存在的时候都会返回错误提示：

|  |
| --- |
|  |

实验总结：

本次实验，总的来说，较为简单，是在Windows系统下去调用相应的API，实现对内存，进程的进行的监视，并且在控制台输出

通过这次实现，我了解到了GetSystemInfo, VirtualQueryEx,VirtualAlloc, GetPerformanceInfo, GlobalMemoryStatusEx ...

等一系列windows 内部API的使用，并且初步了解了Windows 内部对于内存与进程的监控机制，实验过程中也出现了一些问题，比如

在打印进程详细信息的时候一开始返回了空句柄，后来查阅信息知道了是因为权限不足导致。