

**操作系统课程设计**

|  |  |
| --- | --- |
| **实验四、进程控制** | **Experiment 4, Memory Monitoring** |

学院：计算机学院

专业：计算机科学与技术

学生姓名：夏奇拉

学号：1820171025

班级：07111705

Table of Contents

[Purpose 3](#__RefHeading___Toc1766_1221548062)

[Problem Discussion 3](#__RefHeading___Toc1744_1221548062)

[Execution [Windows] 5](#__RefHeading___Toc1768_1221548062)

[Results and Analysis [Windows] 9](#__RefHeading___Toc1770_1221548062)

[Execution [Linux] 10](#__RefHeading___Toc604_992951594)

[Results and Analysis [Linux] 15](#__RefHeading___Toc1770_12215480621)

[Reference: 16](#__RefHeading___Toc606_992951594)

# Purpose

Experiment for Windows:

Windows designs a memory monitor that requires:

Real-time display of the memory usage in the current system, including the layout of the system address space and the usage of physical memory;

Real-time display of the virtual address space layout and working set information of the experiment 2 process control (ParentProcess.exe)

Related syscalls:

GetSystemInfo, VirtualQueryEx, VirtualAlloc, GetPerformanceInfo, GlobalMemoryStatusEx...

Linux experiment:

Use the top command to view the system, subcommands P, T, M

Use ps -A to view all processes and find the pid of ProcessParent

Use top -p pid to check the status of the ProcessParent program;

Use pmap -d pid to view the memory usage of ProcessParent

# Problem Discussion

The experiment requires that the program must display

1. the system’s memory usage
2. the system address space layout
3. the physical memory usage
4. virtual address space
5. parentprocess.exe from lab 2’s working information

**GetSystemInfo**

GetSystemInfo() retrieves information about the current system. It accepts a pointer to a SYSTEM\_INFO structure that receives the information.

The SYTEM\_INFO structure contains information about the current computer system.

* wProcessorArchitecture: The processor architecture of the installed operating system.
* DwPageSize: The page size and the granularity of page protection and commitment. This is the page size used by the VirtualAlloc function.
* lpMinimumApplicationAddress: A pointer to the lowest memory address accessible to applications and dynamic-link libraries (DLLs).
* lpMaximumApplicationAddress: A pointer to the highest memory address accessible to applications and DLLs.
* dwAllocationGranularity: The granularity for the starting address at which virtual memory can be allocated.

**VirtualAlloc**

The VirtualAlloc function reserves, commits, or changes the state of a region of pages in the virtual address space of the calling process. Memory allocated by this function is automatically initialized to zero. It takes 4 arguments.

* LpAddress: the starting address of the region to allocate.

**VirtualAllocEx**

The VirtualAllocEx function is used to allocate memory in the address space of another process.

# Execution [Windows]

|  |
| --- |
|  |

## Results and Analysis [Windows]

# Execution [Linux]

|  |
| --- |
|  |

## Results and Analysis [Linux]

# Reference:

* <https://learn.microsoft.com/en-us/windows/win32/api/sysinfoapi/nf-sysinfoapi-getsysteminfo>
* <https://learn.microsoft.com/en-us/windows/win32/api/sysinfoapi/ns-sysinfoapi-system_info>
* <https://learn.microsoft.com/en-gb/windows/win32/sysinfo/getting-hardware-information?redirectedfrom=MSDN>
* <https://www.installsetupconfig.com/win32programming/windowsvolumeapis1_6.html>
* <https://learn.microsoft.com/en-us/windows/win32/winprog64/virtual-address-space>
* <https://learn.microsoft.com/en-us/windows/win32/memory/memory-management>
* <https://www.tutorialspoint.com/operating_system/os_memory_management.htm>#