# CS 452 Lab 10

# Memory Management under Windows

## Overview

The purpose of this lab is to investigate the operation of the memory management system used in Windows 8/10/11. As with the prior Linux-based virtual memory lab, the main idea is to learn via observation and experimentation. Another goal of this lab is to introduce a number of graphical administrator tools available under Windows; these utilities provide the same information as their command-line Linux counterparts (e.g., vmstat). If you do not have access to a windows machine, you may attempt to access a windows machine remotely: https://winlab.gvsu.edu/

Please note you if you use the above link you will not have admin access and won't be able to complete all of the exercises below, for those that you did not have permission to complete, please make a note in your lab submission.

## Hand-in:

- A word document containing the requested information (answers to the questions)
- Any requested screenshots (uploaded as separate files)
- Program source code (no zip files)

## Windows Memory Management

Most of the concepts covered in the textbook and class (page mapping/translation, virtual memory, working set theory) also appear in the various components that Windows uses to implement memory management. As you step through these experiments, try to relate your observations to concepts mentioned in class.

### Obtaining Memory Information

Just as there are tools in Linux (e.g. **free**, **vmstat**) that report memory parameters and memory usage statistics, there are tools available in Windows that allow us to determine memory status. The Windows-based tools are targeted at administrators; they're also graphical and very informative.

**Perform the following operations and answer the questions:**

The familiar Windows Task Manager has built-in system-monitoring capabilities:

- Start up **Task Manager**: press Ctl-Alt-Del, or right-click **Start | Run...** and enter **taskmgr**
- Click **More details**, then select the **Performance** tab and select **Memory**.

1. What is the *total* amount of Physical Memory (KB) available on your system (In use + Available)?

   Make sure the Task Manager is the only thing running. Then start **Microsoft Edge** and open several tabs (e.g., MSN, Bing, Weather channel) while observing all changes to the **Physical** memory performance metrics. Then close **Edge**.

2. Based on changes to the amount of **Available** memory, what is the apparent footprint (i.e., the full memory demand) of **MS Edge** with several web pages open?
   - Note: this activity is similar to observing process memory demand under Linux using '**vmstat**' (but with lower resolution)

Memory Usage by Competing Programs

It's also possible to observe memory usage using the Performance tool.

**Perform the following operations and answer the questions:**

- Shut-down Task Manager and Edge
- Right-click Start | Computer Management. Open Performance | Monitoring Tools and start up the Performance Monitor tool.
- Click the green Add icon; click on the "Memory" Performance object in the scrollable box; then open the "Memory" Performance object and select the "Available Mbytes" counter from the scrollable list. At this time, you may want to read about this metric by clicking the Show description checkbox to answer the following question:

3. One of the measured components of available memory on Windows is the Standby list. What memory management mechanism described in your textbook does the Windows Standby list implement?
   - Hint: reference section 8.4 of your textbook on Page Replacement concepts and algorithms
   - Ok, since we haven't discussed this yet, Super-Hint: it's on page 394.

- Click on the Add>> button to select this counter, then OK to begin monitoring (plotting) the Available Mbytes.
- To get a better view, right-click in the graph, and choose Properties.
- Click on the Graph tab, and change the Vertical Scale Maximum and the Vertical Scale Minimum fields to appropriate values for the amount of Available memory on your

system. For example, adjust scale to display between 1500 and 2500 MB (based on your observations for questions 1 and 2). You will need to customize this in order to see and measure changes in the graph. Click OK.

- Select (click) a row in the legend at the bottom of the tool to display numerical details about the selected counter. You can also select graph colors and clear the display to more easily track current data.
- Observe (and quantify) the graphed value while starting, running, and terminating two instances of Microsoft Edge, i.e., start one instance, and observe the change in available memory (similar to the vmstat lab). Then start another instance (shift-click), and repeat your observation.
  - Note: hover the mouse pointer over a line in the performance graph to see numeric values for the counter the line represents. You may also want to click the Freeze Display button to pause the display to allow you to observe desired phenomena closely.

4. Note the changes in the reported amount of available memory as graphed in the display. Why is the apparent memory footprint of two instances of MS Edge not exactly twice the memory usage of a single instance?

## Virtual Memory Usage

Like all modern general-purpose operating systems, Windows uses virtual memory. Several tools included with Windows provide a detailed look at virtual memory system parameters.

### Perform the following operations and answer the questions:

- Shut-down the Performance tool and Edge
- Right-click **Start | Run...** and enter **msinfo32** to access the System Information report

5. What are the amounts of Total Physical Memory and Total Virtual Memory available on your system?
   - Explain the relationship between these two numbers and why Total Virtual Memory is bigger.

- Exit System Info
- Open **File Explorer**, Right-click on **This PC**, then select **Properties**
- Click **Advanced system settings**, select the **Advanced** tab, then under **Performance** click the **Settings** button
- Select the **Advanced** tab

6. Under Virtual Memory, observe the size of the **paging file**
   - how does this number correspond with the values observed in question 5?
   - what is the purpose of the paging file in Windows (i.e. what Linux object has similar functionality)?

- Exit the System Settings; leave File Explorer running
- Right-click **Start | Run...** and enter **resmon** to start the Resource Monitor
- Select the **Memory** tab
- Hover over the **Commit (KB)** column heading for an explanation of what it is reporting. Click on any heading to sort it.

7. How much virtual memory is File Explorer (**explorer.exe**) using?

## Page Faults

### Perform the following operations and answer the question:

- Shut-down the Resource Monitor and File Explorer
- Start up an instance of the Windows Accessory **Notepad**.
- Right-click **Start | Computer Management.** Open **Performance | Monitoring Tools** and start up the **Performance Monitor** tool.
- Click the **Add** icon; click on the "Process" **Performance object** in the scrollable box; then open the "Process" **Performance object** and select the "Page Faults/sec" **counter** from the scrollable list.
    - Note: at this time, you may want to read about this metric by clicking the **Show description** checkbox.
- In the scrollable box labeled "Instances of selected objects", find and select **Notepad**; then click on **Add** and then the **OK** button to begin monitoring the selected metric.
- To get a better view, right-click in the graph, and choose **Properties.**
- Click on the **Graph** tab, and change the **Vertical Scale Maximum** to, say, 50. Click **OK**.
- Select (click) a row in the legend at the bottom of the tool to display numerical details about the selected counter. You can also select graph colors and clear the display to more easily track current data.
- Observe the graphed values in their baseline state. Then type some text into Notepad and let the system stabilize. Now observe the graph while changing the font type or asking for ***Bold italic***.

8. Based on your understanding of the concepts discussed in class, what exactly is happening to produce the changes observed in the Performance graph? Quantify your answer.

## Mini-programming Assignment (Memory Management Functions)

### Programming for Windows

This week's programming assignment introduces some very simple program development as a way of becoming familiar with the Microsoft Visual Studio package.

Note: the most recent Microsoft Visual Studio tool suite is available free to all CS students for personal use.

## Program Specifications:

In addition to using performance-monitoring tools, operating systems often provide run-time access to information via an Application Programming Interface (API). For example, it is possible to ask a Windows system to report how big a page is, or to report back on the status of a block of memory. That's precisely the goal of this short programming exercise.

The function prototypes below give the syntax for using the relevant calls. Note: you must include the **<Windows.h>** header file to use these functions.

```
void GetSystemInfo(LPSYSTEM_INFO lpSystemInfo);
SIZE_T VirtualQuery(LPCVOID lpAddress, PMEMORY_BASIC_INFORMATION
lpBuffer, SIZE_T dwLength);
```

These functions follow a familiar O.S. paradigm: perform a system call using the appropriate data structure as a parameter; the operating system fills up the structure with run-time information; this allows your program to then access the fields of the structure to display the desired info. You can read about these functions and their corresponding data structures in the Microsoft Developer Network (MSDN).

Write a Windows-based program that:

1. Queries the system to determine the page size
2. Allocates a large chunk of memory: 1M ($2^{20}$) bytes
3. Queries the system to determine the state of the allocated memory
4. Reports the State in user-friendly terms (i.e., Committed, Reserved, Free)
5. De-allocates the memory
6. Repeats the query to determine/report memory State

Note: please make your program human-friendly (i.e., report the state as readable text, not as a hexadecimal number)

Please upload a screenshot of your program execution.