Anuska Pant
CSCE 611 - OPERATING SYSTEMS

Machine Problem 4: Virtual Memory Management and Memory Allocation

Introduction

The objective of this machine problem was to create a virtual memory manager and virtual memory allocator.

Part I: Support for Large Address Spaces

Previously we implemented direct memory mapping, which is suitable when the number of address spaces and the size of the requested memory is small; otherwise, we very quickly run out of frames in the directly-mapped frame pool. We can make use of "Recursive Table Lookup on the x86" to provide support for logical address spaces when paging is enabled. If we set the last entry in the page directory to point to the page directory itself, we can use it to store the page directory in logical address space.

| 1023: 10 | 1023: 10 | offset: 12 |

Part II: Preparing class PageTable to handle Virtual Memory Pools

We are to add support for registering a virtual memory pool with the page table by checking the legitimacy of a logical address, and freeing pages from memory.

- Added support for registration of virtual memory pools by maintaining a list of registered pools.
- Added support for region check in page fault handler. We check for all the registered pools if the address is legitimate and if it is we proceed with handling page faults else abort the process.
- Added support for releasing previously allocated frames. Page table releases the frame, marks the page invalid and flushes the TLB.

Part III: An Allocator for Virtual Memory

Here we create a lazy allocator which doesn't immediately allocate frames and will simply store the base address and size of the region, which is later allocated when page fault occurs. We allocate regions in multiples of pages. An array is used to store the allocated regions.

This machine problem was a little complicated compared to the others. Calculation of frame number using recursive page table and the associated bit manipulation was a little complicated and it took a lot of time.
The output with the test passed is attached below.