Jacob Hessong

MP4 Design Document

This machine problem focused on completing a few smaller tasks to bring our previous memory managers up to being a full virtual memory manager. First, the ability to perform a recursive page table lookup had to be added to the already created page table. To do this, the function for handling faults had to be updated to recursively use the address in the directory and table so that it may access the fault correctly.

Second, a few more functions had to be added to the page table class for it to work with virtual memory management. The function register_pool was added to maintain a list of pools of memory that have been created. It does this through a list of VMPools of a predefined size, which was added in the PageTable class for this purpose. The number of the VMPools register is also kept in another variable. Next, the page fault handler has to ensure that the address given is valid. This is done using the is_legitimate function in the VMPool class, which will be discussed later. Finally, the free_page function had to be added to the PageTable class. This uses the release_frames function from the ContFramePool class the for the frame number argument. This page table entry is also marked as invalid.

Finally, the VMPool class had to be implemented. The first function that had to be implemented was the constructor. All of the arguments passed into the constructor needed variables to hold them, as well as a new structure that was added to help with allocating regions of memory. Furthermore, a variable to keep track of allocated regions was added to help with allocating and releasing frames. The first page of the VMPool is also used to store the list of regions through the base address. The next function was the allocate function. The first region is given to the page that maintains the VMPool list. For the release function, the base address is determined and is used to locate the region it came from. The number of pages for that region is then found and all of these pages are then released. The region list is then updated to reflect the recently released frames. The final function is the is_legitimate function, which was implemented simply by checking if the address given is with in the minimum (base address) and maximum (base address + offset) limits of the pool.