

Operating Systems

Project 3

Virtual Memory

You are to implement virtual memory mechanism. Here are the key parts of your project:

Main memory. Main memory consists of 100 blocks, each 128 bytes in size.

Virtual memory. Virtual memory is a table of references to main memory. Each item in the table is a pair NUMBER:PHYSICAL ADDRESS. Each NUMBER represents the number of a block in main memory, and PHYSICAL ADDRESS represents the physical address of that block.

Swap memory. Swap memory is a file on a hard disk. It should be 4 times larger than main memory. When there is no enough space in main memory, blocks are swapped into swap memory as needed.

Your program should:

- Allocate a space for main memory dynamically, using malloc
- Generate virtual memory table accordingly.
- Prepare the swap memory.
- Start adding random data (in a loop) to main memory via virtual memory as follows:
 - Reference to a byte as BLOCK NUMBER:OFFSET. For example, to write 42nd byte of 13th block, your data writing function should use address 13:42.
 - Virtual memory should convert this address to physical address using its memory table. It should add the physical address of specified block with the offset.
 - Program should log the time and virtual memory address of allocation.
- When main memory becomes full and another allocation request is made, you program should move the oldest (use the log!) block into swap memory (external file) and continue allocating until swap memory is full. When both main and swap memory are full your program should print out the log and terminate.