

Write-up:

Bit masks and shifts for used to extract the page number and the offset from the virtual address.

- Page: (virtual address & 0xF00) >> 8
- Offset: virtual address & 0xFF

A struct with the following fields called entry was used to model data for the page table and TLB:

- integer page
 - page number in virtual address space
- integer frame
 - frame number in physical address space
- integer valid
 - flag that indicates whether the entry is a valid or not
- long counter
 - counter to track usage for LRU algorithm to replace entries

Data Structures:

- TLB array
 - type entry
 - size 4
- entry table array
 - type entry
 - size 8
- physical memory array
 - type integer
 - size 8 x 64

Algorithm:

1. Read logical address from addresses file.
2. Extract page and offset from virtual address.
3. Check if entry for page exists in TLB.
4. If entry exists in TLB, update counter accordingly for entry to reflect usage by appending a 1 to the binary value of the counter.
5. Skip to step 12 if entry for TLB exists.
6. If entry does not exist in TLB, check if entry for page exists in page table.
7. If entry exists in page table, insert entry into an available empty slot (first-fit placement algorithm) in the TLB. If no slot is empty, then find the counter with the smallest value and replace that entry (least recently used algorithm).
8. Skip to step 11 if entry for page table exists.
9. if entry does not exist in page table, determine the byte offset in BACKING_STORE binary file based on the page, and copy the data into the physical memory array.
10. Insert entries into available slots in (first-fit placement algorithm) page table and TLB. If no slot(s) is/are empty, then find the counter with the smallest value and replace that entry (least recently used algorithm).
11. Update counters for both entries by appending a 1 to the binary value of the counter in the page table and the TLB to reflect usage.
12. Update counters for all entries in TLB and page table by bit shifting counters to right by 1.
13. While there are still addresses not processed yet, repeat from step 1.

