A4 Part 1

Shao Zhang (301571321), Mingyang Cheng (301472892)

1. A certain computer provides its users with a virtual-memory space of 232 bytes. The computer has 218 bytes of physical memory. The virtual memory is implemented by paging, and the page size is 4096 bytes. A user process generates the virtual address 11123456 (this is in hexadecimal (base 16)). Explain how the system establishes the corresponding physical location. Distinguish between software and hardware operations. Feel free to use a diagram (simple ASCII is fine) if you wish, but that's not required.

0x11123456 is 0b 0001 0001 0001 0010 0011 0100 0101 0110
4096 bytes equals to 2^{12} bytes
Therefore,

- Virtual memory = 2^{32} / 2^{12} = 2^{20} pages which is the same as page table size
- Physical memory = 2^{18} / 2^{12} = 2^{6} pages which means physical page number has 6 bits

The first 20 bits "0001 0001 0001 0010 0011" will be the page number which point to a physical page, and last 12 bits "0100 0101 0110" will be the page offset (displacement).

Software Operation:

- When a process generates a virtual address, the OS uses page table to translate this virtual address to a physical address. When the OS manages page table data structure, it maps virtual page numbers to physical page frame numbers which stores the mapping information for each virtual memory page. Page table typically contains the physical address corresponding to a virtual address.
- If a provided physical address does not exist in physical memory, the OS needs to bring the page into memory from the disk (secondary storage). This involves loading the page into an available physical page frame and updating the page table to reflect the new mapping.

Hardware Operation:

- Memory management unit in hardware is responsible for translating virtual address to physical address, includes separate the virtual address into page number and displacement then use page table to find the corresponding physical address.
- If a page fault occurs during the translation process, the hardware triggers an interrupt, and control is transferred to the OS to handle the page fault.

2. Assume we have a demand-paged memory. The page table is held in registers. It takes 8 milliseconds to service a page fault if an empty page is available or if the replaced page is not modified, but 20 milliseconds if the replaced page was modified. Memory access time is 100 nanoseconds. Assume that the page to be replaced is modified 70% of the time. What is the maximum acceptable page-fault rate for an effective access time of no more than 200 nanoseconds? Show your work.

Using the access time formula and plug in the values:
100ns * (1 - p) + (8ms * 0.3 + 20ms * 0,7) = 200 ns

We get:
p = 100 / 1639990 $\approx$ 6.1 * 10^{-6}