Do not publish/post/distribute.

Chapter 9 Main Memory

Questions:

1. Name two differences between logical and physical addresses.
2. Why are page sizes always powers of 2?
3. Consider a logical address space of 64 pages of 1,024 words each, mapped onto a physical memory of 32 frames.
   a. How many bits are there in the logical address?
   b. How many bits are there in the physical address?
4. Explain the difference between internal and external fragmentation.
5. Given six memory partitions of 300 KB, 600 KB, 350 KB, 200 KB, 750 KB, and 125 KB (in order), how would the first-fit, best-fit, and worst-fit algorithms place processes of size 115 KB, 500 KB, 358 KB, 200 KB, and 375 KB (in order)? Rank the algorithms in terms of how efficiently they use memory.
6. Explain why mobile operating systems such as iOS and Android do not support swapping.
7. Assuming a 1-KB page size, what are the page numbers and offsets for the following address references (provided as decimal numbers)?
   a. 308
   b. 42095
   c. 215201
   d. 650000
   e. 2000001
8. The BTV operating system has a 21-bit virtual address, yet on certain embedded devices, it has only a 16-bit physical address. It also has a 2-KB page size.
   a. How many entries are there in a single-level page table?
   b. What is the maximum amount of physical memory?
9. Consider a logical address space of 256 pages with a 4-KB page size, mapped onto a physical memory of 64 frames.
   a. How many bits are required in the logical address?
   b. How many bits are required in the physical address?
10. Consider a computer system with a 32-bit logical address and 4-KB page size. The system supports up to 512 MB of physical memory.
    How many entries are there in a single-level page table?
11. What is the purpose of paging the page tables?
12. On a system with paging, a process cannot access memory that it does not own. Why? (hint: page table)

Solutions:

1. A logical address does not refer to an actual existing address; rather, it refers to an abstract address in an abstract address space. Contrast this with a physical address that refers to an actual physical address in memory. A logical address is generated by

the CPU and is translated into a physical address by the memory management unit (MMU). Therefore, physical addresses are generated by the MMU.

2. Recall that paging is implemented by breaking up an address into a page and offset number. It is most efficient to break the address into X page bits and Y offset bits, rather than perform arithmetic on the address to calculate the page number and offset. Because each bit position represents a power of 2, splitting an address between bits results in a page size that is a power of 2.
3. Consider a logical address space of 64 pages of 1024 words each, mapped onto a physical memory of 32 frames.

    a. Logical address:
        Number of pages = 64 = 2^6
        Words in each page = 1024 = 2^{10}.
        Total entries in the logical memory = $2^6 * 2^{10} = 2^{6+10} = 2^{16}$
        Number of bits = 16

    b. Physical address:
        Number of frames = 32 = 2^5
        Words (entries) in each page = 2^{10}
        Total entries = $2^5 * 2^{10} = 2^{5+10} = 2^{15}$
        Number of bits = 15

6. There are three reasons: First is that these mobile devices typically use flash memory with limited capacity and swapping is avoided because of this space constraint. Second, flash memory can support a limited number of write operations before it becomes less reliable. Lastly, there is typically poor throughput between main memory and flash memory.

7. Assuming a 1-KB page size, what are the page numbers and offset

    Quotient is the page number
    page number = 308 / 1024 = 0

    Remainder is the offset.
    308 mod 1024 = 308

8. The BTV operating system has a 21-bit virtual address, yet on certain embedded devices, it has only a 16-bit physical address. It also has a 2-KB page size.

    Logical address bits = 21
    Page size = 2KB = 2^{11}

    Number of Pages = $2^{21/}2^{11} = 2^{10}$ pages, each page number is in the page table. An entry for every page.

    a. Conventional, single-level page table will have 2^{10} = 1024 entries.
    b. 16-bit physical address can hold a maximum of holds 2^{16} addresses.
        2^{16} = 65536 B = 64 KB

9. Consider a logical address space of 256 pages with a 4-KB page size, mapped onto a physical memory of 64 frames.
   a. Logical address:
      Number of pages = 256 = 2^8
      Page size = 4*1024 = $2^2 * 2^{10} = 2^{12}$
      Total entries = $2^8 * 2^{12} = 2^{8+12} = 2^{20}$
      Number of bits = 20
   b. Physical address:
      Number of frames = 64 = 2^6
      Page size = 2^{12}
      Total entries = $2^6 * 2^{12} = 2^{18}$
      Number of bits = 18
10. An entry for each page. $2^{32} / 2^{12} = 2^{20}$
11. In certain situations, the page tables could become large enough that by paging the page tables, one could simplify the memory allocation problem.