

Main Memory



Practice Exercises

- 8.1 Name two differences between logical and physical addresses.
Answer: 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.
- 8.2 Consider a system in which a program can be separated into two parts: code and data. The CPU knows whether it wants an instruction (instruction fetch) or data (data fetch or store). Therefore, two base-limit register pairs are provided: one for instructions and one for data. The instruction base-limit register pair is automatically read-only, so programs can be shared among different users. Discuss the advantages and disadvantages of this scheme.
Answer: The major advantage of this scheme is that it is an effective mechanism for code and data sharing. For example, only one copy of an editor or a compiler needs to be kept in memory, and this code can be shared by all processes needing access to the editor or compiler code. Another advantage is protection of code against erroneous modification. The only disadvantage is that the code and data must be separated, which is usually adhered to in a compiler-generated code.
- 8.3 Why are page sizes always powers of 2?
Answer: 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.
- 8.4 Consider a logical address space of 64 pages of 1024 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?

Answer:

- a. Logical address: 16 bits
- b. Physical address: 15 bits

- 8.5 What is the effect of allowing two entries in a page table to point to the same page frame in memory? Explain how this effect could be used to decrease the amount of time needed to copy a large amount of memory from one place to another. What effect would updating some byte on the one page have on the other page?

Answer: By allowing two entries in a page table to point to the same page frame in memory, users can share code and data. If the code is reentrant, much memory space can be saved through the shared use of large programs such as text editors, compilers, and database systems. “Copying” large amounts of memory could be effected by having different page tables point to the same memory location.

However, sharing of nonreentrant code or data means that any user having access to the code can modify it and these modifications would be reflected in the other user’s “copy.”

- 8.6 Describe a mechanism by which one segment could belong to the address space of two different processes.

Answer: Since segment tables are a collection of base–limit registers, segments can be shared when entries in the segment table of two different jobs point to the same physical location. The two segment tables must have identical base pointers, and the shared segment number must be the same in the two processes.

- 8.7 Sharing segments among processes without requiring that they have the same segment number is possible in a dynamically linked segmentation system.

- a. Define a system that allows static linking and sharing of segments without requiring that the segment numbers be the same.
- b. Describe a paging scheme that allows pages to be shared without requiring that the page numbers be the same.

Answer: Both of these problems reduce to a program being able to reference both its own code and its data without knowing the segment or page number associated with the address. MULTICS solved this problem by associating four registers with each process. One register had the address of the current program segment, another had a base address for the stack, another had a base address for the global data, and so on. The idea is that all references have to be indirect through a register that maps to the current segment or page number. By changing these registers, the same code can execute for different processes without the same page or segment numbers.

- 8.8 In the IBM/370, memory protection is provided through the use of *keys*. A key is a 4-bit quantity. Each 2K block of memory has a key (the storage key) associated with it. The CPU also has a key (the protection key) associated with it. A store operation is allowed only if both keys are equal, or if either is zero. Which of the following memory-management schemes could be used successfully with this hardware?
- a. Bare machine
 - b. Single-user system
 - c. Multiprogramming with a fixed number of processes
 - d. Multiprogramming with a variable number of processes
 - e. Paging
 - f. Segmentation

Answer:

- a. Protection not necessary, set system key to 0.
- b. Set system key to 0 when in supervisor mode.
- c. Region sizes must be fixed in increments of 2k bytes, allocate key with memory blocks.
- d. Same as above.
- e. Frame sizes must be in increments of 2k bytes, allocate key with pages.
- f. Segment sizes must be in increments of 2k bytes, allocate key with segments.

