8.10 Consider a paging system with the page table stored in memory.
   
a. If a memory reference takes 200 nanoseconds, how long does a 
paged memory reference take?
   
b. If we add TLBs, and 75 percent of all page-table references are found 
in the TLBs, what is the effective memory reference time? (Assume 
that finding a page-table entry in the TLBs takes zero time, if the 
entry is there.)

8.22 Consider a logical address space of 32 pages with 1,024 words per page, 
mapped onto a physical memory of 16 frames.
   
a. How many bits are required in the logical address?
   
b. How many bits are required in the physical address?

9.3 A page-replacement algorithm should minimize the number of page 
faults. We can achieve this minimization by distributing heavily used 
pages evenly over all of memory, rather than having them compete for 
a small number of page frames. We can associate with each page frame 
a counter of the number of pages associated with that frame. Then, 
to replace a page, we can search for the page frame with the smallest 
counter.

   a. Define a page-replacement algorithm using this basic idea. Specifically 
address these problems:
      i. What is the initial value of the counters?
      ii. When are counters increased?
      iii. When are counters decreased?
      iv. How is the page to be replaced selected?
   
b. How many page faults occur for your algorithm for the following 
reference string with four page frames?

       1, 2, 3, 4, 5, 3, 4, 1, 6, 7, 8, 7, 8, 9, 7, 8, 9, 5, 4, 5, 4, 2.

   c. What is the minimum number of page faults for an optimal page-
replacement strategy for the reference string in part b with four 
page frames?

9.6 Consider a demand-paging system with a paging disk that has an 
average access and transfer time of 20 milliseconds. Addresses are 
translated through a page table in main memory, with an access time of 1 
microsecond per memory access. Thus, each memory reference through 
the page table takes two accesses. To improve this time, we have added 
an associative memory that reduces access time to one memory reference 
if the page-table entry is in the associative memory.

   Assume that 80 percent of the accesses are in the associative memory 
and that, of those remaining, 10 percent (or 2 percent of the total) cause 
page faults. What is the effective memory access time?

9.31 Consider the parameter $\Delta$ used to define the working-set window in 
the working-set model. What is the effect of setting $\Delta$ to a small value 
on the page-fault frequency and the number of active (nonsuspended) 
processes currently executing in the system? What is the effect when $\Delta$ 
is set to a very high value?

9.33 Suppose that your replacement policy (in a paged system) is to examine 
each page regularly and to discard that page if it has not been used since 
the last examination. What would you gain and what would you lose 
by using this policy rather than LRU or second-chance replacement?