CSE 421/521 – Operating Systems
Fall 2013 - Homework Assignment #3

The homework assignments will **not** be graded. Please solve these problems this week as an exercise for self-assessment. You may expect similar questions in quizzes and in the exams. The solutions will be provided next week for your convenience.

Problem 1:
Explain the difference between internal and external fragmentation. Suggest ways to prevent each.

Problem 2:
Given five memory partitions of 100 KB, 500 KB, 200 KB, 300 KB, and 600 KB (in order), how would each of the first-fit, best-fit, and worst-fit algorithms place processes of 212 KB, 417 KB, 112 KB, and 426 KB (in order)? Which algorithm makes the most efficient use of memory?

Problem 3:
Compare paging with segmentation with respect to the amount of memory required by the address translation structures in order to convert virtual addresses to physical addresses.

Problem 4:
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 associative registers, and 75 percent of all page-table references are found in the associative registers, what is the effective memory reference time? (Assume that finding a page-table entry in the associative registers takes zero time, if the entry is there.)

Problem 5:
Why are segmentation and paging sometimes combined into one scheme?

Problem 6:
A certain computer provides its users with a virtual-memory space of 2^{32} bytes. The computer has 2^{18} 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. Explain how the system establishes the corresponding physical location. Distinguish between software and hardware operations.

Problem 7:
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 the replaced page is not modified, and 20 milliseconds if the replaced page is modified. Memory access time

is 100 nanoseconds. Assume that the page to be replaced is modified 70 percent of the time. What is the maximum acceptable page-fault rate for an effective access time of no more than 200 nanoseconds?

Problem 8:
Discuss situations under which the least frequently used (LFU) page-replacement algorithm generates fewer page faults than the least recently used page replacement (LRU) algorithm. Also discuss under what circumstance does the opposite holds.

Problem 9:
Consider a demand-paging system with the following time-measured utilizations:
- CPU utilization 20%
- Paging disk 97.7%
- Other I/O devices 5%

Which (if any) of the following will (probably) improve CPU utilization? Explain your answer.
   a. Install a faster CPU.
   b. Install a bigger paging disk.
   c. Increase the degree of multiprogramming.
   d. Decrease the degree of multiprogramming.
   e. Install more main memory.
   f. Install a faster hard disk or multiple controllers with multiple
   a) hard disks.
   g. Add prepaging to the page fetch algorithms.
   h. Increase the page size.

Problem 10:
A page-replacement algorithm should minimize the number of page faults. We can do 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 that are associated with that frame. Then, to replace a page, we search for the page frame with the smallest counter.

a. Define a page-replacement algorithm using this basic idea. Specifically address the problems of (1) what the initial value of the counters is, (2) when counters are increased, (3) when counters are decreased, and (4) how the page to be replaced is selected.

b. How many page faults occur for your algorithm for the following reference string, for 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?