Homework for Week 3
(Due Date: Check eCampus)

1. A computer system has a 36-bit virtual address space with a page size of 8K, and 4 bytes per page table entry.
   (a) How many pages are in the virtual address space?
   (b) What is the maximum size of addressable physical memory in this system?
   (c) If the average process size is 8GB, would you use a one-level, two-level, or three-level page table? Why?
   (d) Compute the average size of a page table in question 3 above.

2. In a 32-bit machine we subdivide the virtual address into 4 segments as follows:

10-bit	8-bit	6-bit	8-bit

   We use a 3-level page table, such that the first 10-bit are for the first level and so on.
   (a) What is the page size in such a system?
   (b) What is the size of a page table for a process that has 256K of memory starting at address 0?
   (c) What is the size of a page table for a process that has a code segment of 48K starting at address 0x1000000, a data segment of 600K starting at address 0x80000000 and a stack segment of 64K starting at address 0xf0000000 and growing upward (like in the PA-RISC of HP)?

3. In a 32-bit machine we subdivide the virtual address into 4 pieces as follows:

8-bit	4-bit	8-bit	12-bit

   We use a 3-level page table, such that the first 8 bits are for the first level and so on. Physical addresses are 44 bits and there are 4 protection bits per page. Answer the following questions, showing all the steps you take to reach the answer. A simple number will not receive any credit.
   (a) What is the page size in such a system? Explain your answer (a number without justification will not get any credit).
   (b) How much memory is consumed by the page table and wasted by internal fragmentation for a process that has 64K of memory starting at address 0?
   (c) How much memory is consumed by the page table and wasted by internal fragmentation for a process that has a code segment of 48K starting at address 0x1000000, a data segment of 600K starting at address 0x80000000 and a stack segment of 64K starting at address 0xf0000000 and growing upward (towards higher addresses)?

4. A computer system has a page size of 1,024 bytes and maintains the page table for each process in main memory. The overhead required for doing a lookup in the page table is 500 ns. To reduce this overhead, the computer has a TLB that caches 32 virtual pages to physical frame mappings. A TLB lookup requires 100ns. What TLB hit-rate is required to ensure an average virtual address translation time of 200ns?

References

[1] A. Silberschatz, P. Galvin, and G. Gagne, *Applied Operating Systems Concepts*, John Wiley & Sons, Inc., New York, NY, 2000.

[2] Deitel, Deitel, and Choffnes, *Operating Systems*, Pearson / Prentice Hall, 2004.

[3] A. S. Tanenbaum, *Modern Operating Systems*, Pearson / Prentice Hall, 2008.

[4] L. F. Bic, A. C. Shaw, *Operating Systems Principles*, Prentice Hall 2003.

[5] C. Crowley, *Operating Systems, A Design-Oriented Approach*, Irwin 1997.

[6] M. Herlihy, N. Shavit, *The Art of Multiprocessor Programming*, Elsevier, 2008