The Objective. The goal of this assignment is to make sure you have a good understanding of materials in Chapters 7 and 8 of the textbook. You should read relevant chapters and check the slides to answer the questions.

Questions. If you have a difficulty with any question, please contact me, TA or post a question on Piazza. Do not post answers on Piazza. If you want to check something of the form 'am I on right track with this question', post it as a PRIVATE question on Piazza. TA/I will make it public whenever appropriate.

How to submit the assignment. Submit your answers using pdf files via Handin. You can find the link on course’s website.

1. (8 pts) 1 Mbyte block of memory is allocated using the buddy system as follows:
   - Request A: 35 Kbytes
   - Request B: 85 Kbytes
   - Request C: 60 Kbytes
   - Request D: 45 Kbytes
   - Release B
   - Release A
   - Request E: 124 Kbytes
   - Release C
   - Release D
   - Request F: 70 Kbyte

Show the status of memory for the mentioned sequence above step by step (similar to Figure 7.6 in the textbook).

2. (5 pts) Consider a simple paging system with the following parameters: $2^{32}$ bytes of physical memory; page size of $2^{10}$ bytes, $2^{16}$ pages of logical address space.
   a. How many bits are in a logical address?
   b. How many bytes are in a frame?
c. How many bits in the physical address specify the frame?

d. How many entries in the page table?

e. How many bits in each page table entry? Assume each page table entry contains a valid/invalid bit.

3. (5 pts) Write the binary translation of the logical address 0001 0100 1011 1010 under the following hypothetical memory management scheme and explain your answer:

A paging system with a 256-address space page size, using a page table in which the frame number happens to be four times smaller than the page number.

4. (6 pts) Consider the memory system with

Virtual memory = 32 GB
Physical memory = 8 GB
Page size = 512 Bytes
Number of processes = 128

Identify
- Size of each page table entry in bits (Assume P/M bits and 4 other bits)
- Size of page table entry in bytes (if it is either 1, 2, 4, 8, 16 bytes etc)
- Number of entries in the page table
- Page table size per process
- Total page table size

5. (6 pts) Solve the same problem with 2-level paging. Assume that the number of bits for identifying level 1 page is 8. In particular

Identify
- Size of each page table entry in bits (Assume P/M bits and 4 other bits)
- Size of page table entry in bytes (if it is either 1, 2, 4, 8, 16 bytes etc)
- Number of entries in the page table for level 1 page table
- Number of entries in the page table for level 2 page table
- Page table size per process
- Total page table size

6. (4 pts) Solve the same problem for inverted page table. Assume that the number of bits for each inverted page table entry is 32 bytes. In particular

Identify
- Number of entries in the page table
- Page table size per process
- Total page table size

7. (16 pts) Consider the following memory accesses: Assume 3 frames Identify number of page faults under

1. OPT
2. LRU
3. FIFO
4. Clock

3 2 4 2 4 2 1 3 4 4 5 2 2 4 2 5 5 2 2 1