CS/ECE 752 ADVANCED COMPUTER ARCHITECTURE I

HOMEWORK # 4 (Due at lecture on Nov 6th, Fri)

1. (9 points)
   Consider a virtual memory system with the following properties:

- 44 bit virtual address (byte addressable)
- 4 kB pages
- 40 bit physical addresses (byte addressable)
- 1 MB cache that is 2-way set-associative and has a line size of 64 bytes, and is accessed with physical addresses
- a 64 entry TLB

a. What is the total size of the page table for each process on this machine, assuming that the valid, protection, dirty, and use bits take a total of 4 bits, and that all of the virtual pages are in use? (Assume that disk addresses are not stored in the page table).
b. Why might it be infeasible to represent a page table as in (a)? Hierarchical page tables have multiple levels of page tables (segregated by groups of bits in the virtual address). Do hierarchical page tables resolve the issue?
c. Draw a diagram of the hardware in the memory system including the cache and TLB. Make sure you show how different fields of the address (i.e. which bits) are used to access the cache and TLB. Use Figure C.24 in H&P (page C-47) as a model for your diagram.

2. (9 points)
   Problem 5.1 a, b, and c in H&P (page 343)

3. (6 points)
   Problem 5.2 a and c in H&P (page 343)

4. (4 points)
   Consider the following piece of code that transposes an integer matrix a and stores the result as the integer matrix b.

```
for (int i = 0; i < 64; i++)
  {
    for (int j = 0; j < 64; j++)
     {
     b[j][i] = a[i][j]
     }

   }
```

Assume that this is executed on a system with the following cache:

- fully associative
- size 64kB
- no pre-fetching
- block size of 32 bytes
- write allocate and write back

Assume that the size of an integer is 4 bytes.

a. Determine the number of conflict, compulsory, and capacity misses while executing this piece of code. Do this separately for both read and write misses.

A "simple pre-fetcher" is defined as follows:
If cache misses, fetch the missed block and the next one block.
If cache hits, no memory fetch.

b. Determine the number of conflict, compulsory, and capacity misses if a simple pre-fetch unit is added to the system. Do this separately for both read and write misses.