Cpr E 381 Homework 11

1. Problem 7.11 Given the following pseudocode:

   ```
   int array[10000,100000];

   for each element array[i][j]{
        array[i][j] = array[i][j] * 2;
   }
   ```

   Write two C programs that implement this algorithm: one should access the elements of the array in row-major order, and the other should access them in column-major order. Compare the execution times of the two programs. What does his tell you about the effects of memory layout on cache performance?

2. Problem 7.28 Associativity usually improves the miss ratio, but not always. Give a short series of address references for which a two-way set-associative cache with LRU replacement would experience more misses than a direct-mapped cache of the same size. (Be clear in your demonstration).

3. Problem 7.32 Consider three processors with different cache configurations:
   - Cache 1: Direct-mapped with one-word blocks
   - Cache 2: Direct-mapped with four-word blocks
   - Cache 3: Two-way set associative with four-word blocks

   The following miss rate measurements have been made:
   - Cache 1: Instruction miss rate is 4%; data miss rate is 6%.
   - Cache 2: Instruction miss rate is 2%; data miss rate is 4%.
   - Cache 3: Instruction miss rate is 2%; data miss rate is 3%.

   For these processors, one-half of the instructions contain a data reference. Assume that the cache miss penalty is 6 + Block size in words. The CPI for this workload was measured on a processor with Cache 1 and was found to be 2.0. Determine which processor spends the most cycles on cache misses.

4. Problem 7.35 The following C program is run (with no optimizations) on a processor with a cache that has eight-word (32-byte) blocks and holds 256 bytes of data:

   ```
        int i, j, c, stride, array[512];
        …
        for(i=0; i<10000; i++)
             for(j=0; j<512; j=j+stride)
                  c = array[j]+17;
   ```

   If we consider only the cache activity generated by references to the array and we assume that integers are words, what is the expected miss rate when the cache is direct mapped and stride = 256? How about if the stride = 255? Would either of these change if the cache were two-way set associative?

5. Problem 7.39 Consider a virtual memory system with the following properties:
   - 40-bit virtual byte address
   - 16 KB pages
   - 36-bit physical byte address

   What is the total size of the page table for each process on theis processor, assuming that the valid, protection, dirty, and use bits take a total of 4 bits and that all virtual pages are in use? (Assume that disk addresses are not stored in the page table.)
6. Problem 7.41 A processor has a 16-entry TLB and uses 4KB pages. What are the performance consequences of this memory system if a program accesses at least 2 MB of memory at a time? Can anything be done to improve performance?

7. Problem 7.52 Why might a compiler perform the following optimization?

```
/* Before */
for(j=0; j<20; j++)
     for(i=0; i<200; i++)
          x[i][j] = x[i][j] + 1;

/* After */
for(i=0; i<200; i++)
     for(j=0; j<20; j++)
          x[i][j] = x[i][j] + 1;
```