Brown University
School of Engineering
EN1640 Design of Computing Systems
Professor Sherief Reda
HW 05. 24 points. Due Date: Wednesday April 27th in class.

1. (4 points) You are building an instruction cache for an ARM processor. It has a total capacity of $4C = 2^{c+2}$ bytes. It is $N=2^n$-way set associate ($N \geq 8$), with a block size of $b=2^{b'}$ bytes ($b \geq 8$). Give your answers to the following questions in terms of these parameters.
    a. (1 point) Which bits of the address are used to select a word within a block?
    b. (1 point) Which bits of the address are used to select the set within the cache?
    c. (1 point) How many bits are in each tag?
    d. (1 point) How many tag bits are in the entire cache?

2. (4 points) We discussed the least recently used (LRU) replacement policy for multiway associative caches. Other, less common, replacement policies include first-in-first-out (FIFO) and randon policies. FIFO replacement evicts the block that has been there for the longest, regardless of how recently was accessed. Random replacement randomly picks a block to evict
    a. (2 points) Discuss the advantages and disadvantages of each of these replacement policies.
    b. (2 points) Describe a data access pattern for which FIFO would perform better than LRU.

3. (4 points) Suppose you are running a program that is attempting to load words from the following addresses. The address pattern is executed only once.

    0x0, 0x8, 0x10, 0x18, 0x20, 0x28

    a. (1 points) If you use direct mapped cache with cache size 1 KB and a block size of 8 bytes (two words), how many sets (i.e., lines) are in the cache?

    b. (1 points) With the same cache and block size as in part (a), what is the miss rate of the direct mapped cache for the given memory access pattern?

    c. (2 points) For the given memory access pattern, which of the following would decrease the miss rate the most? (Cache capacity is kept constant). Circle one
        a. Increasing the degree of associativity to 2
        b. Increasing the block size to 16 bytes
        c. Either (i) or (ii)
        d. Neither (i) nor (ii)

4. (12 points) Most scientific applications involve matrix operations. The most common operation is matrix multiply: X=YZ, where X, Y, and Z are *N*-by-N matrices. Assume that the elements of X are computed row-by-row, and matrix sizes are 256 × 256. Each element is a double precision floating-point number (8 bytes). Assume a fully associative data cache with a least recently used (LRU) replacement policy. Assume a total cache size of 128 KB and that the cache block size is 8 bytes as well and that operands are aligned.

a. [6 points] Compute the total number of misses in the data cache. To compute the data cache miss rate, we neglect all integer access in the program execution and focus on the floating-point accesses. Also compute the total number of operations (FP ADDs and FP Mult).

b. [6 points] To reduce the miss rate, it is possible to block the matrix multiply into operations on sub-matrices of size *N/k* by *N/k* (*k* is power of 2). The miss rate and the execution time should improve because several submatrices easily fit in the cache now. Describe the block algorithm that should be implemented to minimize the number of misses, and compute the overall number of misses. Also compute the total number of operations (FP ADDs and FP Mult)