CSCI 2330 – Cache Design Exercises

Suppose you have a program that uses **100 bytes of memory** (addresses 0 to 99), and makes the memory accesses specified below, in this order (assume that each access is for a single byte only):

a. **READ 27**
b. **WRITE 64**
c. **READ 83**
d. **WRITE 92**
e. **WRITE 45**
f. **READ 41**
g. **READ 55**
h. **WRITE 43**

Consider a hypothetical cache that **can only hold a single block at a time, initialized to block 0**. For each of the following possible configurations of this 1-block cache, determine whether each memory access in the list above would be a cache **HIT** or a cache **MISS**. Also determine how many **memory trips** are needed by each access. Provide the total number of hits, misses, and memory trips for each configuration.

A good approach is to trace through each configuration one access at a time, keeping track of which block is currently held in the cache and whether the block is clean or dirty.

1. 50-byte blocks, write-through, no-write-allocate

2. 25-byte blocks, write-through, no-write-allocate

3. 50-byte blocks, write-back, write-allocate (assume that write-misses load into the cache first and then modify the cached block only)

4. 25-byte blocks, write-back, write-allocate (assume that write-misses load into the cache first and then modify the cached block only)