# Homework 4 Solutions

**Problem 1**

| Reference | Hit or Miss |
|---|---|
| 1 | Miss |
| 4 | Miss |
| 8 | Miss |
| 5 | Miss |
| 20 | Miss |
| 17 | Miss |
| 19 | Miss |
| 56 | Miss |
| 9 | Miss |
| 11 | Miss |
| 4 | Miss |
| 43 | Miss |
| 5 | Hit |
| 6 | Miss |
| 9 | Hit |
| 17 | Hit |

Here is the final state of the cache:

| Block # | Address |
|---|---|
| 0 | |
| 1 | 17 |
| 2 | |
| 3 | 19 |
| 4 | 4 |
| 5 | 5 |
| 6 | 6 |
| 7 | |
| 8 | 56 |
| 9 | 9 |
| 10 | |
| 11 | 43 |
| 12 | |
| 13 | |
| 14 | |
| 15 | |

## Problem 2

| Reference | Hit or Miss |
|---|---|
| 1 | Miss |
| 4 | Miss |
| 8 | Miss |
| 5 | Hit |
| 20 | Miss |
| 17 | Miss |
| 19 | Hit |
| 56 | Miss |
| 9 | Miss |
| 11 | Hit |
| 4 | Miss |
| 43 | Miss |
| 5 | Hit |
| 6 | Hit |
| 9 | Miss |
| 17 | Hit |

Here is the final state of the cache:

| Block # | Address |
|---|---|
| 0 | 16 |
| 1 | 4 |
| 2 | 8 |
| 3 | |

## Problem 3

AMAT = Hit time + Miss time × Miss penalty.
AMAT = 2 ns + (20 × 2 ns) × 0.05 = 4 ns.

## Problem 4

AMAT = (1.2 x 2 ns) + (20 × 2 ns × 0.03) = 2.4 ns + 1.2 ns = 3.6 ns.
Yes, this is a good choice.

## Problem 5

Execution time = Clock cycle × IC × (CPI + Cache miss cycles per instruction)
Execution $\text{time}_{\text{original}}$ = Clock cycle × IC × (CPI + Cache miss cycles per instruction)
Execution $\text{time}_{\text{original}} = 2 \times \text{IC} \times (2 + 1.5 \times 20 \times 0.05) = 7\ \text{IC}$
Execution $\text{time}_{\text{new}} = 2.4 \times \text{IC} \times (2 + 1.5 \times 20 \times 0.03) = 6.96\ \text{IC}$

Hence doubling the cache size to improve miss rate at the expense of stretching the clock cycle results in essentially no net gain.

## Problem 6

The largest direct-mapped cache with one-word blocks would be 256 KB in size. The address breakdown is 14 bits for tag, 16 bits for index, 0 bits for block offset, and 2 bits for byte offset. A total of 12 chips will be required — 4 for overhead.

## Problem 7

If the block size is four words, we can build a 512-KB cache. The address breakdown is 13 bits for tag, 15 bits for index, 2 bits for block offset, and 2 bits for byte offset. A total of 18 chips will be required — 2 for overhead.

## Problem 8

Here are the cycles spent for each cache:

| Cache | Miss Penalty | Instruction Miss Cycle per Instruction | Data Miss Cycle per Data Reference | Total Miss Cycles per Instruction |
|---|---|---|---|---|
| C1 | 6 + 1 = 7 | 4% × 7 = 0.28 | 8% × 7 = 0.56 | 0.56 |
| C2 | 6 + 4 = 10 | 2% × 10 = 0.20 | 5% × 10 = 0.50 | 0.45 |
| C3 | 6 + 4 = 10 | 2% × 10 = 0.20 | 4% × 10 = 0.40 | 0.40 |

So, C3 spends the least time on misses and C1 spends the most.

## Problem 9

Execution time = CPI × Clock cycle × Instruction count
Execution $\text{time}_{\text{C1}} = 0.56 \times 2\ \text{ns} \times \text{IC} = 1.12 \times \text{IC} \times 10^{-9}$
Execution $\text{time}_{\text{C2}} = 0.45 \times 2\ \text{ns} \times \text{IC} = 0.9 \times \text{IC} \times 10^{-9}$
Execution $\text{time}_{\text{C3}} = 0.40 \times 2.4\ \text{ns} \times \text{IC} = 0.96 \times \text{IC} \times 10^{-9}$

So C2 is the fastest and C1 is the slowest.