CSE4010-02 Computer Architecture Final Exam

---

1. Please fill the blank with O if the statement is true. Otherwise fill it with X.

(a) Pipelining reduces cycle time not CPI. ( )

(b) The pipelined MIPS processor reduces the execution time over the unpipelined processor. ( )

(c) Forwarding is used to reduce the pipeline stall due to structural hazard. ( )

(d) Full (complete) forwarding can remove all data hazards in 5-stage pipelined MIPS processors. ( )

(e) A deep pipeline (e.g. more pipeline stages) can increase CPI. ( )

(f) A bigger cache block when a cache size is fixed can increase conflict misses.( )

(g) A write-through cache requires the larger memory bandwidth than a write-back cache. ( )

(h) Bank-interleaving in the main memory increases the bandwidth of the memory system. ( )

(i) Instruction cache can exploit both temporal and spatial locality. ( )

(j) Fully associative cache do not need index bits ( )

2. Floating point arithmetic [15 points]

(a) **[5 points]** Suppose a machine whose floating number format is as follow.

sign: 1 bit
exponent: 3 bit (bias = 3)
significand 4 bit

Represent the following two numbers with this format
1.3125
0.375

(b) [5 points] Add above two floating point numbers and represent the result in the floating point format.

(c) **[5 points]** In above floating point format, what is the smallest positive denormalized number? Express it in both float point format and decimal number. You may just show the equation in case of decimal number.

3. Pipelined processor [15 points]

Consider the instruction block given below where $t1, $t2, and $t4 all point to different memory addresses:

```
addi $t1,  $t10, -10
lw $t2, 0($t1)
lw $t3, 0($t2)
sw $t3, 0($t4)
sub $t3, $zero, $t3
addi $t3, $t3, 1
```

(a) **[8 points]** Using the same code block as in part (a), show each of the stages of execution through a pipelined processor that provides full forwarding and hazard detection logic. Show all required forwarding by drawing an arrow from the source to the destination. Indicate pipeline stalls by a dashed line (--). The first one has been done for you.

Instruction	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th	11th	12th	13th	14th
addi $t1 $t0 -10	IF	ID/RF	EX	MEM	WB									
lw $t2 0($t1)														
lw $t3 0($t2)														
sw $t3 0($t4)														
sub $t3 $zero $t3														
addi $t3 $t3 1														

(b) **[7 points]** Below is a slightly different block of code including branch and jump instructions.

```
            beq $t1, $zero, ERROR
            lw $t2, 0($t1)
            add $t3, $zero, $zero
            j END
ERROR:      addi $t3, $zero -1
END:        addi $t0, $t0, 4
```

Consider the case where **$t1 value is 0**. Show the stages of execution through a pipelined processor that uses full forwarding with hazard detection logic and implements a **not-taken branch prediction strategy**. The processor will simply flush the remaining pipeline stages when conditional or unconditional branches are taken. You should also assume that sufficient hardware has been added such that the branch execution (branch condition evaluation and target address calculation) occurs in the ID stage.

Show all required forwarding by drawing an arrow as in part (b). Also, indicate pipeline stalls/flushed stages by a dashed line (--) as in part (b). The first one has been done for you.

Instruction	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th	11th	12th	13th	14th
beq $t1 $0 ERROR	IF	ID/RF	EX	MEM	WB									

4. **Cache [25 points]**

Consider the following C code.

```
int a[100], b[107], i;

for (i=0; i<100; i++)
  a[i] = b[i] + b[i+7];
```

Here are the assumptions.

- Array a is located at 0x100 and array b is located at 0x800.
- The processor has a two-way set associative data cache whose size is 8 blocks and the size of the block is 8 bytes.
- The miss penalty is 100 clock cycles.
- The address is 32 bits.
- The replacement policy is LRU.
- A cache is a writ-through cache.

(a) **[2 points]** Compute the number of bits for the tag, index, block offset.

(b) **[2 points]** What is the memory access order of a[i],b[i],b[i+7]?

(c) **[6 points]** How many misses do you get for the first two iterations?

(d) **[5 points]** How many bytes are written to the memory when you execute the above program? Note that a whole cache block is written to the memory upon a memory write.

Consider following cache configuration.

- The processor has separate instruction and data cache whose size is 32 KB each
- The size of the block is 32 bytes.
- The miss penalty (fetching a block from the memory or writing a block to the memory) is 100 clock cycles.
- A cache is a writeback cache.
- At any given time, 50% of blocks are dirty (written).
- Instruction cache miss rate is 5%
- Data cache miss rate is 3%.
- The percentage of load and store instructions is 20 %.
- No write buffer is used.

(e) **[5 points]** Compute the additional CPI stall cycles due to I cache misses.

(f) **[5 points]** Compute the additional CPI stall cycles due to D cache misses.