CSE4010-01 Computer Architecture and Logic Final Exam

1. [10points] Please fill the blank with O if the statement is true. Otherwise fill it with X.

a. Energy consumption is reduced if we reduce the clock frequency. ( )

b. Two CPUs with the same ISA can have different CPIs. ( )

c. Jump instruction can jump in both forward and backward direction. ( )

d. Pipelined CPU increases CPI over unpipelined CPU. ( )

e. Forwarding in pipelined MIPS processor can completely remove stall cycles. ( )

f. Forwarding can be used to reduce the pipeline stall due to control hazard. ( )

g. Set-associative cache reduces the capacity misses. ( )

h. Increasing set-associativity reduces TAG bits when the size of cache is fixed. ( )

f. A bigger cache block increases the miss penalty. ( )

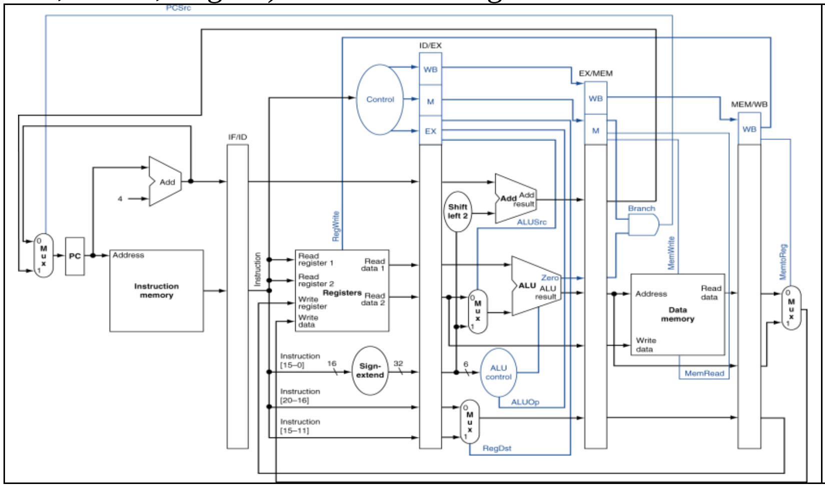
h. Increasing cache block size can take advantage of temporal locality. ( )

2. [10points] Please compute the average CPI. Assume CPI_{base}=1.

(a) [5points] Branch instructions account for 20 % of instructions. Branch resolution and target address calculation are done in ID stage of the MIPS processor. No branch prediction is done. Consider only the control hazard.

(b)[5points] LW/SW accounts for 30 % of instructions. Miss rate for I-cache and D-cache is 2% and 5% and miss penalty is 100 cycles. (Hint: Ignore write-backs and replacement). Consider only the stall due to cache miss. Do not include the stalls in (a).

3. [15points] Please write down the value of control signals (RegWrite, MemRead, MemWrite, PCSrc, RegDst) for the following instructions.



ORI $1, $2, 100
BNE $1, $2, 200
SW $1, $2, 100

4. Pipelined processor [35 points]

Please consider following codes

```
Loop:  lw $2, 0($1)
       lw $3,0($2)
       add  $4, $3, $2
       sw $4, 4($2)
       beq $4, $5, Loop
```

(a)[5 points] Identify all read-after-write dependencies in the code above assuming the code. You should draw a circle around both registers that exhibit the dependency and connect them with an arrow.

(b)[10 points] Assume the pipelined processor provides full forwarding but no hazard detection logic. Assume branch target address calculation and resolution is done in ID and the last beq instruction is taken. Please insert "nop" between instructions and after beq if needed to avoid the hazards. [hint: you can insert two nops if you need two stall cycles.]

(c)[10points] To reduce the clock cycle time, MEM stage is pipelined into two stages *(e.g. IF ID EX MEM WB -> IF ID EX MEM1 MEM2 WB).* How many nops are added to (b) to avoid the hazards. Consider the assumptions in (b). Show your work.

(d)[10 points] This problem is not related to (a)-(c). Assume 50 % of branches are taken. We develop a branch predictor that predicts the outcome (taken, not-taken) of 90 % branches accurately. Compute the additional CPI for this predictor due to control hazards assuming 15 % of instructions are branches and both branch resolution and target computation are done in ID.

5. Cache [30 points]

Consider the following C code.

```
int a[100],b[100];
for (i=0;i<5;i++)
  a[i] = b[i]+b[i+4];
```

Suppose that array a is located at address 0x200 and array b is located at address 0x800. The processor has a data cache whose size is 1 KB and the size of the cache block is 32 bytes.

(a) [5 points] How many bits are allocated for the tag, index, and block offset of the cache assuming we have a 32-bit memory address (assume block offset includes byte offset)?

(b) [5 points] How many load instructions are generated per iteration? How many store instructions are generated per iteration?

(c)[7 points] How many data cache misses when a direct mapped cache is used? Show your work. (No credit for just an answer.)

(d)[8 points] How many data cache misses when a two-way set associative cache is used? Show your work. (No credit for just an answer.)

(e)[5 points] Assuming we use a write-back cache. How many bytes are written back to main memory due to dirty block replacement in (c) ?