CSc/CpE 142 – Exam 2

Name: [redacted]

1) Fill in the blanks /Short answer (30 points, 3 points each)

a) List two primary methods for enhancing instruction-level parallelism.

I. Deeper Pipeline

II. Multi-issue Pipeline

b) Block is the unit of information transferred between the cache and main memory on a miss.

c) Write through is a scheme (policy) in which writes always update both cache and the next lower level of memory hierarchy, ensuring that the data is always consistent between the two.

d) Temporal locality is the principle stating that if a data location is referenced, then it will tend to be referenced again soon.

e) Pipeline -3 is a method of resolving a branch hazard that assumes a given outcome for the branch and proceeds based on that assumption rather than waiting for the actual outcome.

f) Increasing block size -3 can improve hit ratio due to temporal locality.

g) In a direct-mapped cache each block has only one word. True or False. (False circled)

h) The five stages in a pipeline implementation require 4, 5, 4, 2, and , 8 nanoseconds. What is the fastest clock rate at which this datapath can operate?....$\frac{8}{8\times 5} = \frac{1}{8}$ = 0.125 nanoseconds $\frac{5}{8\times 5}$ = 0.125

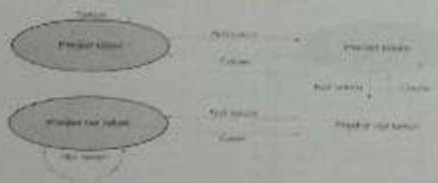
i) Consider a 2-bit branch prediction scheme illustrated by the following figure. What is the accuracy of this *2-bit predictor* for the following repeating pattern of branch outcomes assuming the predictor starts in the top left state? ......8/11, 3 miss

T, NT, NT, NT, NT, NT, NT, T, T, NT

h M h h h h h m h h   -3

h - hit

M - Miss



j) List three types of hazards that can degrade the performance of a pipelined datapath.

i) Control hazard

ii) structural hazard

iii) Data hazard

CSc/CpE 142 – Exam 2

Name: [redacted]

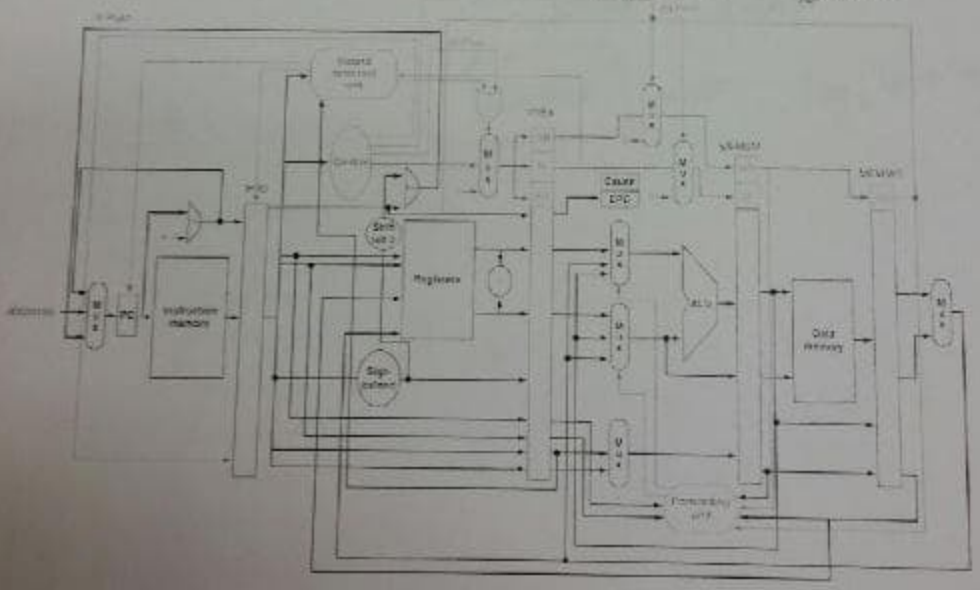
2) Consider the pipelined system shown below. Find the number of cycles required to execute the following MIPS code assuming that branch is taken. Justify your result. (20 points)

		1	2	3	4	5	6	7	8	9	10	11	12	13
	SLT $6, $8, $7	IF	ID	Ex	M	WB								
	LW $5, 100($6)		IF	ID	Ex	M	WB							
	LW $7, 100($15)			IF	ID	Ex	M	WB						
	BEQ $8, $15, L1				IF	ID	Ex	M	WB					
	SUB $9, $7, $4													
	SUB $20, $9, $7													
L1:	LW $8, 100($9)						IF	ID	Ex	M	WB			
	ADD $11, $8, $12							IF	ID	ID	Ex	M	WB	
	SW $8, 104($9)									IF	ID	Ex	M	WB

$S+N-1+(\text{delays}) = 5+7-1+(1+1) = 13$ cycles

Branch hazard

lw hazard



CSe/CpE 142 – Exam 2 Name [redacted]

3) A computer uses 16-bit addresses, with byte addressable locations. Assume this system has a cache containing 32 **(thirty two) 8-word** blocks. Each word is 16 bits long. Show the implementation of the cache assuming **2-way set associativity**.

- You need to show how a given 16-bit address is divided into different parts to perform a cache reference. Show all necessary components of the cache. Provide the size of each component and the widths of its inputs and outputs, if applicable. **(15 points)**
- Assume the cache is empty. Find the number of misses for the following hexadecimal address sequence: 0, 5, 4, 7, 20, 21, 23, 8, C, FC. **(10 points)**

byte offset → $\frac{16\text{ bit}}{\text{word}} \frac{1\text{ byte}}{8\text{ bits}} = \frac{2\text{ byte}}{\text{word}}$ $\log_2 2 = 1$ bit

word offset → $\log_2 8 = 3$ bits

$16 \frac{32\text{ blk}}{\text{cache}} \frac{\text{set}}{2\text{ blk}} = 16\ \text{set/cache}$

Index → $\log_2 16 = 4$ bits

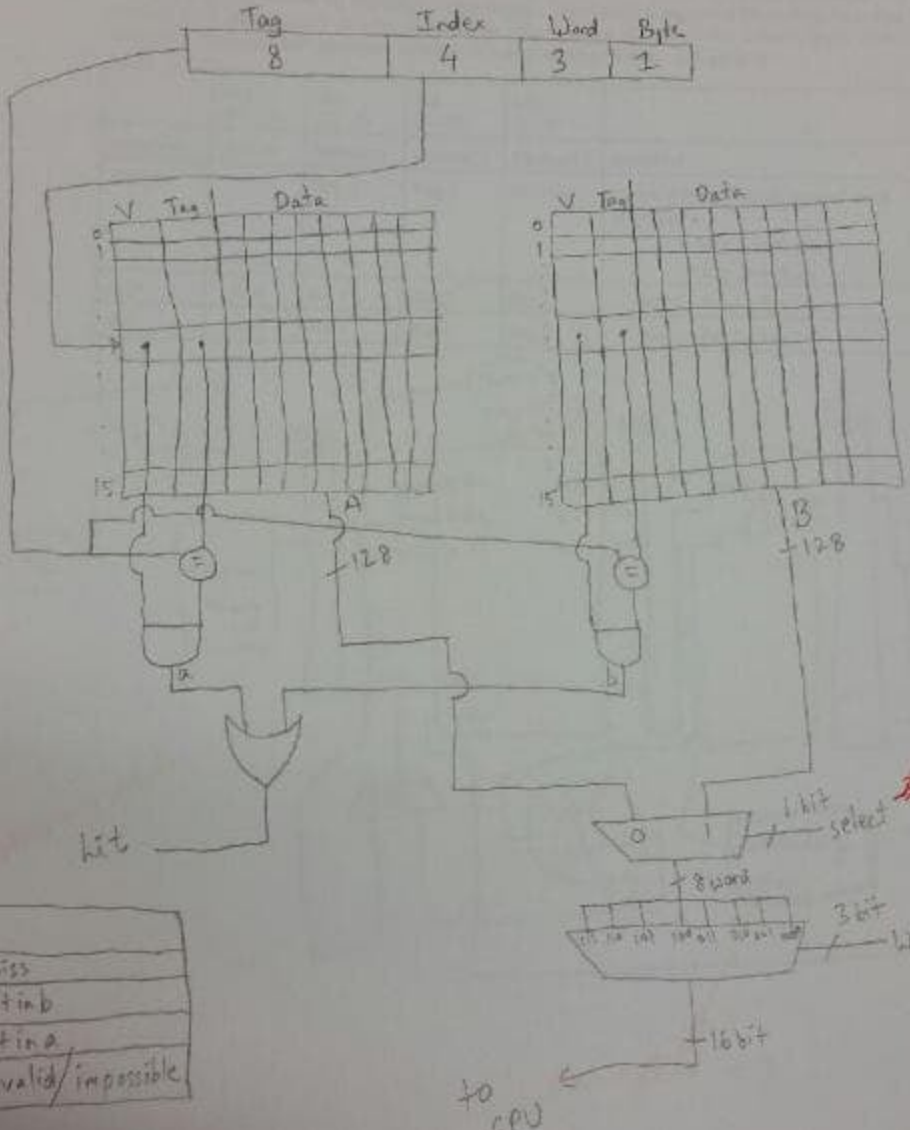
(0-15, 16-31, 32-47)

Dec	hex	
0	0	miss
5	5	hit
4	4	hit
7	7	hit
32	20	miss
33	21	hit
35	23	hit
8	8	hit
12	C	hit
252	FC	miss

why? :3

3 misses

Name: [redacted]



a	b	
0	0	miss
0	1	hit in b
1	0	hit in a
1	1	invalid/impossible

CSe/CpE 142 – Exam 2 Name [redacted]

4) Consider a hypothetical 16-bit assembly language with only three instructions. The format for these instructions is given below. Design a pipelined CPU which can fetch and execute any of these instructions. Show all necessary components. Label each component (including its ports). Assume that the memory is byte-addressable. The internal design of the control logic is not required. You must use minimum number of components possible. (25 points)

	Bits (15 - 12)	bits (11 - 8)	bits (7 - 4)	bits (3 - 0)	
Instruction	opcode	operand 1	operand 2	Operand 3	operation
SWAP	0000	Reg 1	Reg 2	not used	Swaps the contents of operand 1 and operand2; Reg 1 <= Reg 2 Reg 2 <= Reg1
ADD	0010	Reg 1	Reg 2	Reg 3	Reg 1 = Reg 3 + Reg 2
SUB	1000	Reg 1	Reg 2	Reg 3	Reg 1 = Reg 3 - Reg 2