



# University of Asia Pacific

## Admit Card

Final-Term Examination of Spring, 2021

Financial Clearance

PAID

Registration No : 18201043

Student Name : Shawon Barman

Program : Bachelor of Science in Computer Science and Engineering



SI.NO.	COURSE CODE	COURSE TITLE	CR.HR.	EXAM. SCHEDULE
1	CSE 313	Numerical Methods	3.00	
2	CSE 314	Numerical Methods Lab	0.75	
3	CSE 315	Peripheral & Interfacing	3.00	
4	CSE 316	Peripheral & Interfacing Lab	1.50	
5	CSE 317	Computer Architecture	3.00	
6	CSE 319	Computer Networks	3.00	
7	CSE 320	Computer Networks Lab	1.50	
8	CSE 321	Software Engineering	3.00	
9	CSE 322	Software Engineering Lab	0.75	

Total Credit: 19.50

1. Examinees are not allowed to enter the examination hall after 30 minutes of commencement of examination for mid semester examinations and 60 minutes for semester final examinations.
2. No examinees shall be allowed to submit their answer scripts before 50% of the allocated time of examination has elapsed.
3. No examinees would be allowed to go to washroom within the first 60 minutes of final examinations.
4. No student will be allowed to carry any books, bags, extra paper or cellular phone or objectionable items/increditing paper in the examination hall.  
Violators will be subjects to disciplinary action.

Final Semester Examination, Spring - 2021

Name : Shawon Barman

Registration number : 18201093

Year : 3rd

Semester : 2nd

Course code : CSE 317

Course Title : Computer Architecture

Section : A

Exam date : 21-11-2021

Q1. No QW Ans

My registration ID = 18201093

$$\therefore x = 93$$

Given that,

$$A[x+10] = Y + A[x+15];$$

$$\Rightarrow A[93+10] = Y + A[93+15];$$

$$\Rightarrow A[53] = Y + A[56];$$

Let's consider, A has ULA of 16 bits and registers of 16 bits.

$$A = \$500 \quad (1)$$

$$Y = \$51 \quad (2)$$

$$T32 \quad (\text{Temporary register})$$

$$\text{temp} = \$t_0 \quad (3)$$

MIPS machine code:

①  $lw \$t_0, 232(\$s_0);$

②  $add \$t_0, \$s_1, \$t_0;$

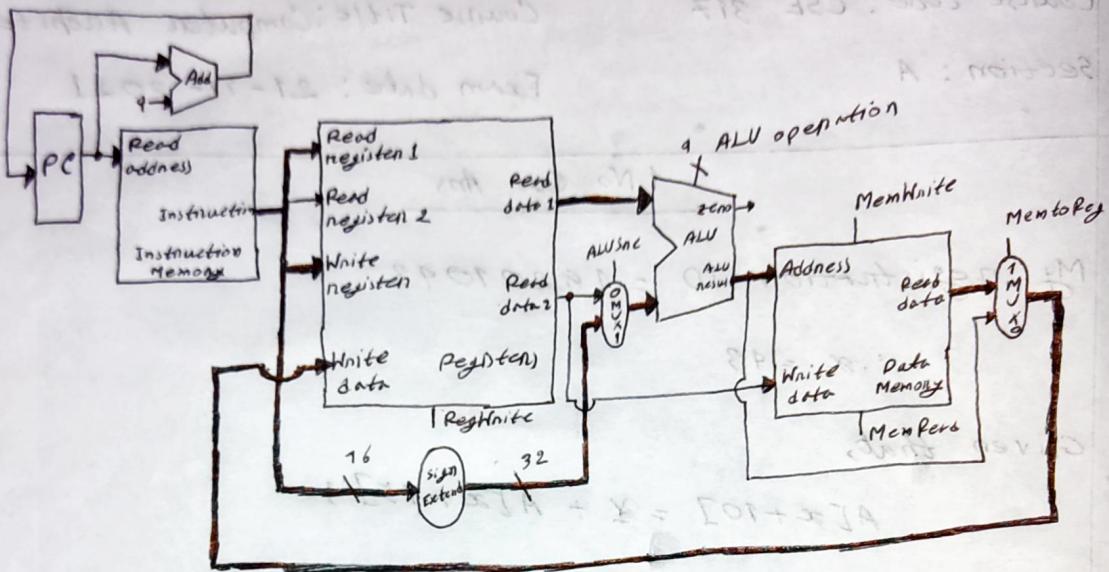
③  $sw \$t_0, 212(\$s_0);$

$$T_1: 56 \times 9 = 232$$

$$T_2: 53 \times 9 = 212$$

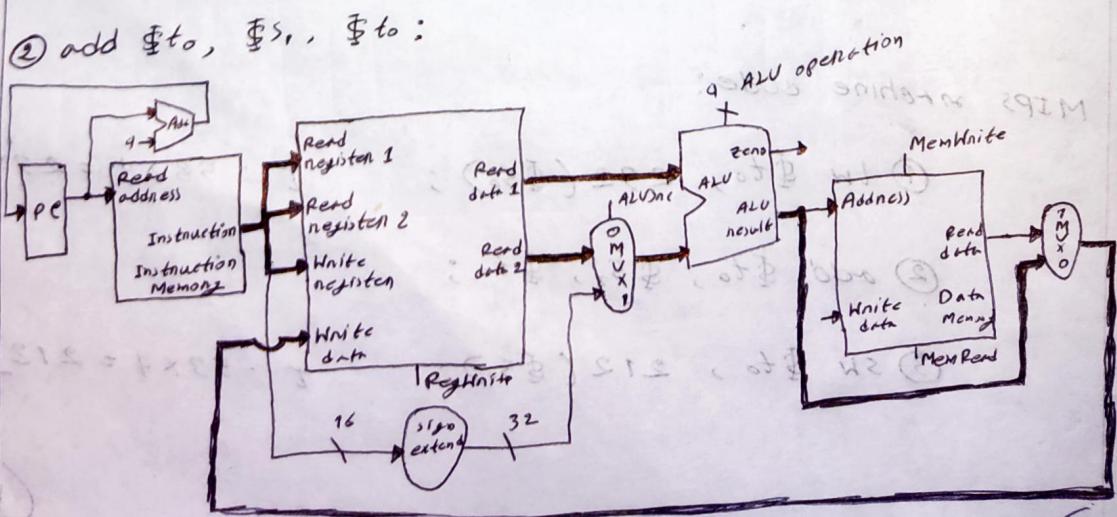
(P.T.O)

① I/W \$to, 232 (\$So):



This is I-type instruction, so ALUSrc is SET and number of that is \$So is passed to read register 1. And we'll get its content that is the base address from Read data 1. 16-bit offset is converted to 32-bit using sign extend block address and offset will go to ALU and ALU will add them thus we'll get the physical address from ALU result. This physical address goes to data memory fetch data from that address and pass that data to write register and RegWrite is SET.

② add \$to, \$S., \$to:



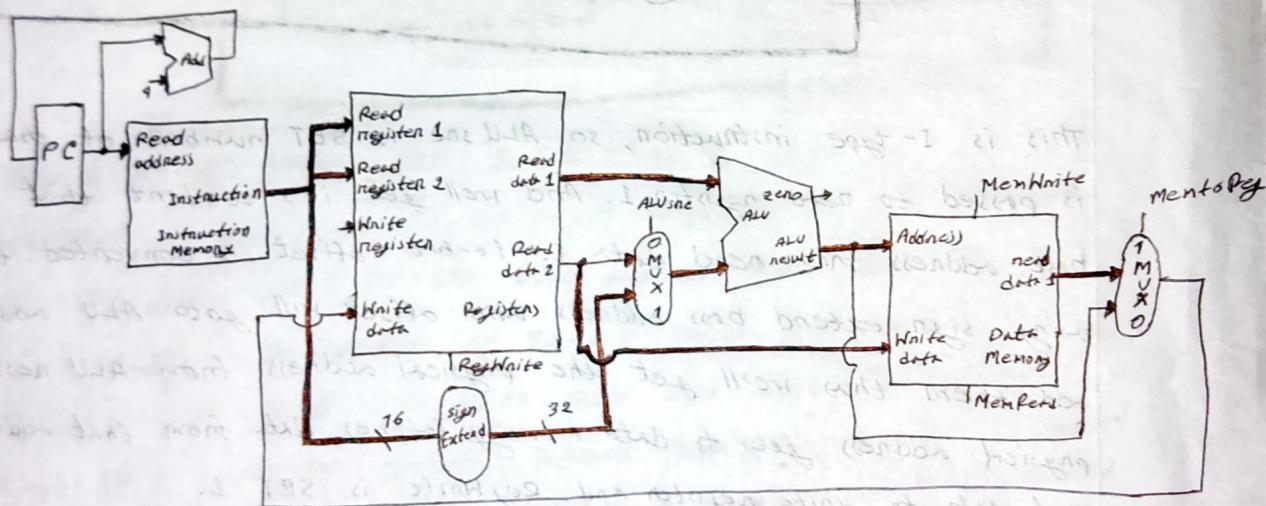
Page 3

Name: SHANON BANNAN

Prg. ID: 18201093

This instruction is R-type instruction, so ALUSinc is CLEAR. The number of \$to and \$S, is 8 and 17 is passed to read register 1 and read register 2 then content of the two from read data 1 and read data 2. These two data is passed to ALU. ALU adds them thus we get the result from ALU. The result is passed to MemtoReg MUX but as it is clear so the MUX passed the result to write of \$S0 that is passed to write register and RegWrite is SET.

③ SW \$to, ~~17~~<sup>212</sup> (\$S0):



As this is an I-type instruction so ALUSinc is SET. The number of \$S0 that is 16 passed to read register 1 and the number of \$to that is 8 is passed to read register 2. So the content of data that is base address from read data 1 and content of \$S0 that is data to be stored from data 2. 16-bit offset is converted to 32-bit. Base address and offset is passed to ALU thus we get physical from ALU result. This physical address and data to be stored is passed to data memory. This is the procedure of this instruction.

Name : Sharon Barman

Reg ID: 18201093

Page 4

Name : Sharm  
Reg ID: 18201093

2 No Ques Ans @

Here, My registration ID: 18201093

$$\therefore x = 93$$

$$\begin{aligned} \text{So, Block address: } & (x+11), (x+21), (x+17) \\ & = (93+11), (93+21), (93+17) \\ & = 59, 69, 60 \end{aligned}$$

No. of blocks in cache = 16 one-word blocks

I. Direct mapped:

<u>Block Address</u>	<u>Cache Block</u>
59	$(59 \% 16) = 6$
69	$(69 \% 16) = 0$
60	$(60 \% 16) = 12$

Request Address	hit / miss	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
59	miss																
69	miss	MEM[6A]															
60	miss	MEM[69]												MEM[60]			

So,

no. of hit = 0

no. of miss = 3

(P.T.O)

Name: Sharon Barman

Reg. ID: 18201093

① 9-Way, 8-Way, and 16-Way (Full associative) mapping:

For 9-Way,

$$\frac{\text{no. of block}}{9} = \frac{76}{9} = 4 \text{ set}$$

<u>Memory block address</u>	<u>cache block</u>
59	$59 \div 9 = 2$
61	$61 \div 9 = 0$
60	$60 \div 9 = 0$

<u>request address</u>	<u>hit on miss</u>	<u>set 0</u>	<u>set 1</u>	<u>set 2</u>	<u>set 3</u>
59	miss				MEM[59]
61	miss	MEM[61]			MEM[59]
60	miss	MEM[60], MEM[61]			MEM[54]

For 8-Way;

$$\frac{\text{no. of block}}{8} = \frac{76}{8} = 2 \text{ set}$$

<u>Memory block address</u>	<u>cache block</u>
59	$59 \div 2 = 0$
61	$61 \div 2 = 0$
60	$60 \div 2 = 0$

<u>request address</u>	<u>hit on miss</u>	<u>set 0</u>	<u>set 1</u>
59	miss	MEM[59]	
61	miss	MEM[59], MEM[61]	
60	miss	MEM[59], MEM[61], MEM[60]	

(T.T.D)

Name: Sharon Barman

Reg. ID: 18201093

page 6

For 72-way

$$\frac{\text{No. of block}}{72} = \frac{14}{16} = 1$$

Memory block address

54

61

60

Cache block

$$54 \% 7 = 0$$

$$61 \% 7 = 0$$

$$60 \% 7 = 0$$

request address	hit or miss	set 0							
		0	1	2	3	4	5	...	75
54	miss	MEM[54]							
61	miss	MEM[61]							
60	miss	MEM[60]							

Scalable cache

0.0848

0.0544

0.0544

Scalable cache

0.0

1.0

0.0

Scalable cache	Scalable cache	Scalable cache	Scalable cache
0.0848	0.0544	0.0544	0.0
0.0544	0.0	1.0	0.0
0.0544	0.0	0.0	0.0

Name: Sharron Barman

Pg. ID: 18209093

2 No Ques Ans (b)

i) Given,

$$\text{Cache size} = 32 \text{ word}$$

$$\text{memory size} = 512 \text{ word}$$

$$\text{block size} = 4 \text{ word}$$

$$\therefore \text{memory block} = \frac{\text{memory size}}{\text{block size}} = \frac{512}{4} = 128 \text{ block}$$

$$\therefore \text{cache line} = \frac{\text{Cache size}}{\text{block size}} = \frac{32}{4} = 8 \text{ block}$$

Hence, total bit for physical address =  $\log_2(\text{memory size})$

$$= \log_2(512)$$

$$= 9 \text{ bit}$$

$$\text{so, index} = \log_2(\text{cache line})$$

$$= \log_2(8)$$

$$= 3 \text{ bit}$$

$$\text{offset} = \log_2(\text{block size}) = \log_2(4) = 2 \text{ bit}$$

$$\text{tag bit} = 9 - 3 - 2 = 4 \text{ bit.}$$

tag	index	offset
4 bit	3 bit	2 bit

ii) Given, cache size = 16 word

$$\text{memory size} = 64 \text{ word}$$

$$\text{block size} = 4 \text{ word}$$

$$\therefore \text{memory block} = \frac{\text{memory size}}{\text{block size}} = \frac{64}{4} = 16 \text{ block}$$

$$\therefore \text{cache line} = \frac{\text{Cache size}}{\text{block size}} = \frac{16}{4} = 4 \text{ block}$$

Name: Shawon Barman

Reg. No: 18201093

Hence, total bit =  $\log_2(64) = 6$  bit

so, index =  $\log_2(4) = 2$  bit

offset =  $\log_2(4) = 2$  bit

tag =  $6-2-2 = 2$  bit

tag	index	offset
2-bit	2-bit	2-bit

$$\text{blocks} = \frac{32}{4} = \frac{32 \times 32}{32 \times 4} = 8 \text{ blocks}$$

(i) Given,

cache size = 16 word

main memory size = 128 word

block size = 4 word

∴ memory block =  $\frac{128}{4} = 32$  blocks

∴ cache line =  $\frac{16}{4} = 4$  blocks

Hence, total bit =  $\log_2(128) = 7$  bits

so, index =  $\log_2(4) = 2$  bit

tag	index	offset
3-bit	2-bit	2-bit

tag =  $7-2-2 = 3$  bit

tag	index	offset
3-bit	2-bit	2-bit

$$\text{blocks} = \frac{32}{4} = \frac{32 \times 32}{32 \times 4} = 8 \text{ blocks}$$

$$\text{blocks} = \frac{16}{4} = \frac{16 \times 32}{16 \times 4} = 4 \text{ blocks}$$

$$\text{blocks} = \frac{4}{1} = \frac{4 \times 32}{4 \times 1} = 32 \text{ blocks}$$

(P.T.O)

(Q. No. 9)

Given,  
cache size = 32 word

memory size = 1024 word

block size = 4 word

∴ memory block =  $\frac{1024}{4} = 256$  block∴ cache line =  $\frac{32}{4} = 8$  blockHence, total bits =  $\log_2(1024) = 10$  bitso, index =  $\log_2(8) = 3$  bitsoffset =  $\log_2(4) = 2$  bits

$$\text{tag} = 10 - 3 - 2 = 5 \text{ bit}$$

tag	index	offset
5-bit	3 bit	2 bit

(Q. No. 9)  
Given, cache size = 8 word

memory size = 16 word

block size = 4 word

∴ memory block =  $\frac{16}{4} = 4$  block∴ cache line =  $\frac{8}{4} = 2$  blockHence, total bit =  $\log_2(16) = 4$  bitso, index =  $\log_2(2) = 1$  bitoffset =  $\log_2(1) = 0$  bit

$$\text{tag} = 4 - 1 - 0 = 3 \text{ bit}$$

tag	index	offset
3 bit	1 bit	2 bit

(P.T.O.)

Name: Sharon Barman  
Reg. ID: 18201099

Page 10

Answered correctly: Smart  
2020-21: Q1 20

### Q No 8 Ans ②

Difference between pipeline and non-pipeline machine are:

- ① In pipeline machine, multiple instructions are overlapped during execution. On the other hand, in a non-pipeline machine, processes like decoding, fetching, execution and writing memory are merged into a single unit on a single step.
- ② Many instructions are executed at the same time in the pipeline machine. On the other hand, only one instruction is executed at the same time in the non-pipeline machine.

Hence, my ID = 18201093

$$\therefore x = 3$$

so, total instruction in a program =  $x + 9 = 3 + 9 = 12$

Number of stage	I <sub>1</sub>	I <sub>2</sub>	I <sub>3</sub>	I <sub>4</sub>	I <sub>5</sub>	I <sub>6</sub>	I <sub>7</sub>	I <sub>8</sub>	I <sub>9</sub>	I <sub>10</sub>	I <sub>11</sub>	I <sub>12</sub>				
IF	I <sub>1</sub>	I <sub>2</sub>	I <sub>3</sub>	I <sub>4</sub>	I <sub>5</sub>	I <sub>6</sub>	I <sub>7</sub>	I <sub>8</sub>	I <sub>9</sub>	I <sub>10</sub>	I <sub>11</sub>	I <sub>12</sub>				
ID	I <sub>1</sub>	I <sub>2</sub>	I <sub>3</sub>	I <sub>4</sub>	I <sub>5</sub>	I <sub>6</sub>	I <sub>7</sub>	I <sub>8</sub>	I <sub>9</sub>	I <sub>10</sub>	I <sub>11</sub>	I <sub>12</sub>				
EXE		I <sub>1</sub>	I <sub>2</sub>	I <sub>3</sub>	I <sub>4</sub>	I <sub>5</sub>	I <sub>6</sub>	I <sub>7</sub>	I <sub>8</sub>	I <sub>9</sub>	I <sub>10</sub>	I <sub>11</sub>	I <sub>12</sub>			
MEM			I <sub>1</sub>	I <sub>2</sub>	I <sub>3</sub>	I <sub>4</sub>	I <sub>5</sub>	I <sub>6</sub>	I <sub>7</sub>	I <sub>8</sub>	I <sub>9</sub>	I <sub>10</sub>	I <sub>11</sub>	I <sub>12</sub>		
WB				I <sub>1</sub>	I <sub>2</sub>	I <sub>3</sub>	I <sub>4</sub>	I <sub>5</sub>	I <sub>6</sub>	I <sub>7</sub>	I <sub>8</sub>	I <sub>9</sub>	I <sub>10</sub>	I <sub>11</sub>	I <sub>12</sub>	
clock cycle time	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

Figure: Page Table

(P.T.O)

stage	clock	time
stage 3	stage 4	stage 5

Name: Sharmin BINTU

Reg. No: P8201093

Hence,

number of stage,  $k = 5$

number of instruction,  $n = 12$

∴ Total time for non-pipeline  $= K \times n$

$$= 5 \times 12$$

$$= 60 \text{ cc} \quad (\text{Ans.})$$

∴ Total time for pipeline  $= K \times 1 + 1 \times (n-1)$

$$= 5 \times 1 + (12-1)$$

$$= 5 + 11$$

$$= 16 \text{ cc} \quad (\text{Ans.})$$

$$\therefore \text{CPI} = \frac{\text{Total clock cycle}}{\text{Total number of instruction}}$$

$$= \frac{76}{12}$$

$$= 1.33 \approx 1 \quad (\text{Ans.})$$

$$\therefore \text{Speedup} = \frac{\text{Total time for non-pipeline}}{\text{Total time for pipeline}}$$

$$= \frac{60 \text{ cc}}{16 \text{ cc}}$$

$$= 3.75 \quad (\text{Ans.})$$

$$\therefore \text{Efficiency} = \frac{\text{total block in table}}{\text{total used block}}$$

$$= \frac{80}{60}$$

$$= 1.33 \quad (\text{Ans.})$$

(P.T.O)

PHU-12  
Name: Sharwan BNIMON

Reg. ID: 18701093

Ex-010589 102.32

3 No Ques Ans (b)

Hence, my registration ID = PB201093

$$\therefore i = 93$$

$$\text{so, } i+1010 = 93 + 1010 = 1053$$

∴ Instruction latency for non-pipeline =  $(35+90+95+30+20) \text{ ns}$

$$= 170 \text{ ns (Ans)}$$

∴ time to execute 1053 instruction for non-pipeline.

machine = instruction latency  $\times$  number of instruction

$$= 170 \times 1053 \text{ ns}$$

$$= 179070 \text{ ns (Ans)} \rightarrow \text{Ans} \rightarrow \text{Ans}$$

∴ Instruction latency for pipeline =  $\max\{35, 90, 95, 30, 20\}$

$$(i.e., 95 \text{ ns (Ans)})$$

time for execute 1053 instruction for pipeline

machine = instruction latency  $\times$  number of stage + (number of instruction - 1)  $\times$  instruction latency

$$= 95 \times 5 + (1053-1) \times 95$$

$$(i.e., 88.8 \text{ ns})$$

$$= 225 + 97390$$

$$= 97565 \text{ ns (Ans)}$$

$$(i.e., 88.8 \text{ ns})$$

(P.T.O)

∴ speedup =  $\frac{\text{total time for non-pipeline}}{\text{total time for pipeline}}$

$$= \frac{179010}{97565} \text{ ns}$$

$$= 3.76 \text{ ns} \quad (\text{Ans})$$

3 No Ques Ans (c)

Given,

lw \$s<sub>2</sub>, 0(\$s<sub>1</sub>) [I<sub>1</sub>]

add \$s<sub>3</sub>, \$s<sub>4</sub>, \$s<sub>2</sub> [I<sub>2</sub>]

sub \$s<sub>6</sub>, \$s<sub>2</sub>, \$s<sub>3</sub> [I<sub>3</sub>]

	I <sub>1</sub>	I <sub>2</sub>	I <sub>3</sub>	
lw				IF
add				MEM
sub				WB

For pipeline,

IF	I <sub>1</sub>	I <sub>2</sub>	I <sub>3</sub>	
ld		I <sub>1</sub>	I <sub>2</sub>	I <sub>3</sub>
EXE			I <sub>1</sub>	I <sub>2</sub>
MEM				I <sub>3</sub>
WB				

Hence, when lw instruction come in IF stage, the add and sub instruction still waiting. When the lw instruction come in LD-stage, the add instruction come in IF stage and sub instruction still waiting. When the lw instruction come in EXE stage, the add instruction come in LD stage and sub instruction come in IF stage. After clock cycle, #5, the lw instruction finished, after clock cycle 6, the add instruction finished, after clock cycle 7, the sub instruction finished.

(P.T.O)

INC-11

Name: Shikha Banerjee

Prg ID: 18201043

BRANCH AND JUMP

PROFESSOR: DR. RAJ

There would be hazard for \$S\_2\$ of instruction 2 is loaded to pipeline before instruction 3 stores new values to \$S\_2\$. Same goes for instruction 3.

For non-pipeline:

(Memory in IF, ID, EXE, MEM, WB)

IF	I <sub>1</sub> , S <sub>2</sub> , I <sub>3</sub>
ID	I <sub>1</sub> , S <sub>2</sub> , I <sub>3</sub>
EXE	I <sub>1</sub> , S <sub>2</sub> , (I <sub>1</sub> , I <sub>2</sub> , I <sub>3</sub> )
MEM	I <sub>1</sub> , S <sub>2</sub> , I <sub>3</sub>
WB	I <sub>1</sub> , I <sub>2</sub> , I <sub>3</sub>

In ~~pipeline~~ non-pipeline, every instruction can happen at the same time because it will complete each instruction at a time and then execute another instruction.

R1	S1	I1	91
R2	S2	I2	92
R3	S3	I3	93

Let's see what happens if we store intermediate value and then write back intermediate value with below line. Intermediate value does not change if we do not write back intermediate value. So, if we do not write back intermediate value, then it will be stored in memory. If we write back intermediate value, then it will be stored in memory. (P.T.O.)

(C.J.Y.)

Name: SHAWN BHIMM

Reg. ID: 18201093

P.T.O. '15

QUESTION PAPER

EXAM NOV/DEC 2018

q No Ans OR ①

Four question about cache design are:

① Where can a block be placed in upper level? (Block placement → mapping function)

② How is a block found if it is in the upper level?  
(Block identification)

③ Which block should be replaced on a miss? (Block replacement).

(a)  $\text{addr} = A$

④ What happens on a write? (Write strategy).

(a)  $\text{addr} = S$

(b)  $\text{addr} = \text{ghost}$

(c)  $\text{addr} = \text{old w1}$  ①

(d)  $\text{addr} = \text{old block w1}$  ②

(e)  $\text{addr} = \text{old due w1}$  ③

(f)  $\text{addr} = \text{old w2 w2}$  ④

110001 = w1  
just moving

110101 = w2

010001 = due

000001 = busy

(P.T.O.)

(Q.59)

Nono: Shweta Bantwal  
Reg. No: 18201093

P.T.O - 16

Nono: Shweta Bantwal  
Reg. No: 18201093

Q No Qn Ans No b

Hence, my registration number = 18201093

Given that,

$$A[i] = B + A[i+1] - C \quad (1)$$

(constant stepwise function)

Where,

$$i = 93$$

so,

$$A[93] = B + A[59] - C; \quad (\text{constant stepwise function})$$

Let's consider,

$$A = \$S_0 \quad (16) \quad (\text{initial value})$$

$$B = \$S_1 \quad (17) \quad (\text{constant value})$$

$$C = \$S_2 \quad (18) \quad (\text{constant value})$$

$$\text{temp} = \$t_0 \quad (8)$$

so,

$$\textcircled{i} \text{ } l_w \$t_0, 216(\$S_0)$$

$$\textcircled{ii} \text{ add } \$t_0, \$t_0, \$S_1$$

$$\textcircled{iii} \text{ sub } \$t_0, \$t_0, \$S_2$$

$$\textcircled{iv} \text{ sw } \$t_0, 172(\$S_0)$$

Given that,

$$l_w = 100011$$

$$s_w = 101011$$

$$\text{sub} = 100010$$

$$\text{add} = 100000$$

(P.T.O)

Page 17

Name: Sharan Bannan

Dg. ID: 16701013

So, the MIPS machine code:

(i)

100011	10000	01000	0000 0000	11011000
op	rs	rt		offset

(ii)

000000	01000	10001	01000	00000	100000
op	rs	rt	rd	shamt	funct

(iii)

000000	01000	10010	01000	00000	100010
op	rs	rt	rd	shamt	funct

(iv)

101011	10000	01000	0000 0000	10101100
op	rs	rt		offset

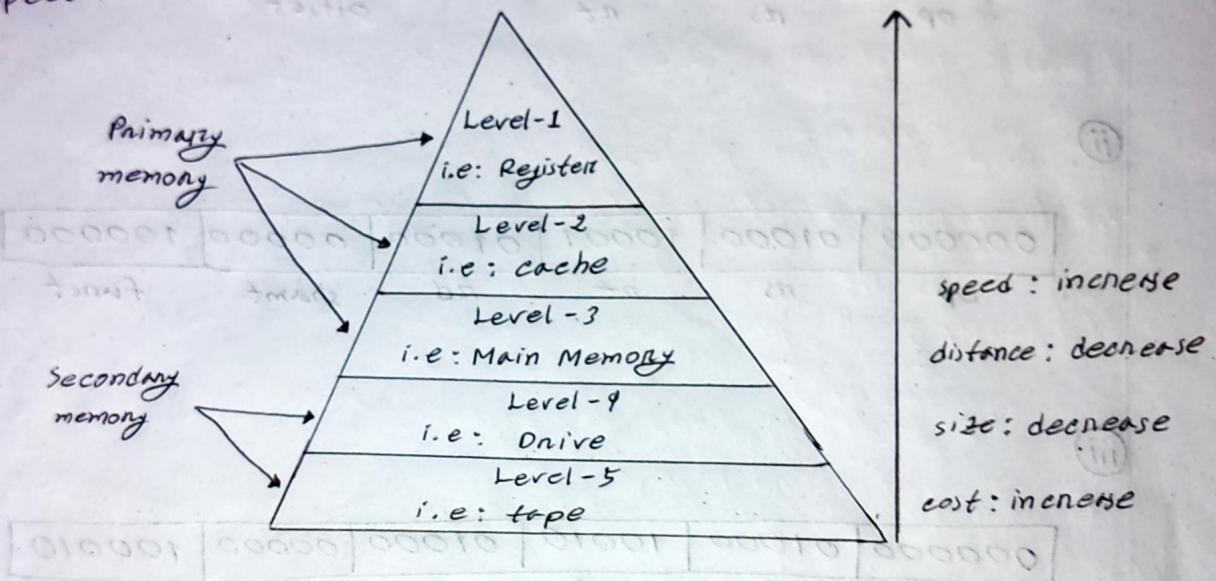
(P.T.O)

Name: SHARON DALLMAN  
Reg. No: 18207043

RHC-18

Q No Ques Ans of ①

Memory Hierarchy according to cost, size, distance, and speed:



In computer architecture the memory hierarchy separates computer storage into a hierarchy based on response time. Since the response time, cost, size, distance, and speed are related the levels may also be distinguished by their performance and controlling technologies. From the above diagram, we can see that when we go to upper level from lower level the size and from factor decrease, distance also decrease, thus increase the speed and cost.