BITS Pilani, Hyderabad Campus

Comprehensive Exam First Semester 2020-21

Computer Architecture (CS F342) Date: 16-December-2020(AN)

Time: 2 hrs Mode: Open Book

Note: This question paper contains three sections. Section B is MCQ type with a negative marking of 25% for each wrong answer. Section A and B is to be answered in google form. Section C is to be answered in a separate answer sheet.

Section-A

Fill in the blanks with **ONE WORD ONLY.** [10X1=10]

1. The memory which is used to store the copy of data or instructions stored in larger memories, inside the CPU is called _____. [Registers]

2. The primary reason to partition a memory into data and instruction memory is resolve _____ hazard. [structural]

3. The effectiveness of the cache memory is based on the property of _____. [locality of reference]

4. The bit used to signify that the cache location is updated is_____ [dirty bit]

5. The process where a content is directly written into main memory without first transferring the content to the cache under write miss is called _____ [No-write allocate]

6. In _____ addressing the 26 bits jump address is shifted and concatenated with the upper 4 bits of the PC. [pseudo-direct]

7. _____ instruction is used (besides lui) to load a 32-bit immediate value into a register.

8. The value -5.678 when rounded to nearest 2 decimal places and ties away from zero will be _____. [-5.68]

9. The values of control signals ALUSrc, PCSrc, MemRead, and MemToReg for a lw instruction will be _____. [1, 0, 1, 1] (Note: No partial marking)

10. The architecture which has multiple processing elements operating on a single data is called _____. [MISD]

Section B MCQs: [12 marks]

1. A certain processor uses a fully-associative cache size of 16kB. The cache block size is 16 bytes. Assume that the main memory is byte addressable and uses a 32-bit address.

How many bits are required for the Tag and the Index fields respectively in the address generated by the processor? **[2]**

a) 24bits and 0bits
b) 28 bits and 4 bits
c) 24 bits and 4 bits
d) 28bits and 0 bits

2. What instruction(s) must be used to convert beq $s0, $s1, L1 instruction to make a far jump? **[1]**
   a) bne $s0, $s1, L2
   b) bne $s0, $s1, L2 and j L1
   c) beq $s0, $s1, L2 and jr L1
   d) beq $s0, $s1, L2 and j L1
3. Consider the following processor design characteristics: **[2]**
   i) Register-to-register arithmetic operations only
   ii) Fixed-length instruction format
   iii) Non-programmable control unit

   Which of the characteristics above are used in the design of a RISC processor?
   (a) (i) and (ii) only
   (b) (ii) and (iii) only
   (c) (i) and (iii) only
   (d) (i), (ii) and (iii)
4. Assume that there are 4 stages in the pipeline IF, ID, EX and WB. Assume that there are n instructions to be executed and any instruction has a RAW dependency on the previous instruction (except the first instruction). Assuming that there is no data forwarding is used, the number of cycles that will be required to complete execution of all the instructions are: **[4]**
   (a) 4n
   (b) 4+3n
   (c) 4+ 3(n-1)
   (d) 2+3n
5. Consider the unsigned fixed point binary number representation below,
   b7b6b5b4b3.b2b1b0

   where position of the binary point is between b3 and b2. Assume b7 is msb. Which of the decimal numbers listed below cannot be represented exactly in the above representation: **[2]**
   (i) 31.500
   (ii) 0.875
   (iii) 12.100
   (iv) 3.001

(a) (iii), (iv)
(b) (ii), (iii), (iv)
(c) Only (iv)
(d) None of them can be represented

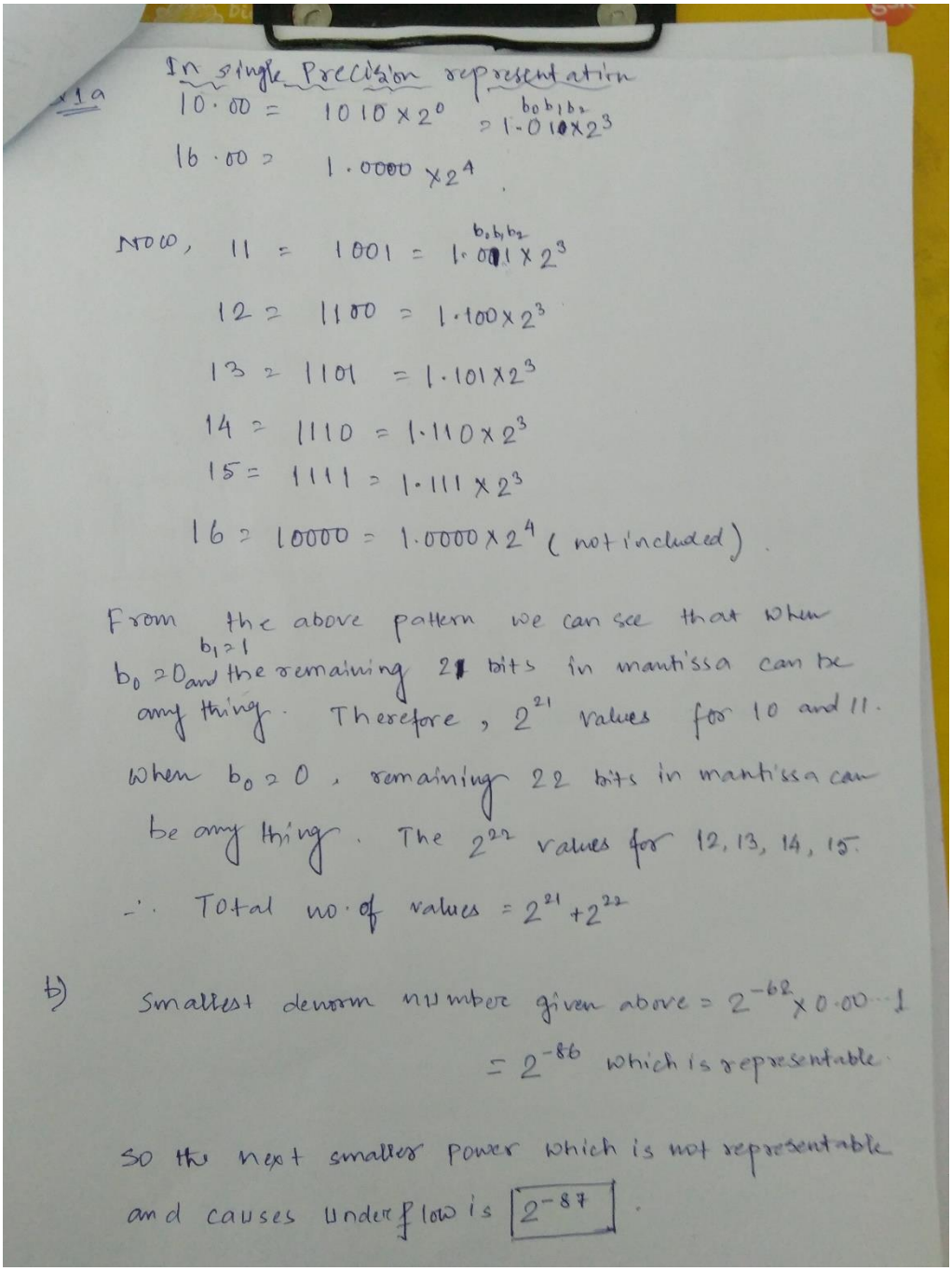
6. Which of the following is/are true for a ‘for’ loop: **[1]**
   (i) it has temporal locality of reference
   (ii) it has spatial locality of reference
   (iii) has 90% prediction accuracy when 1-bit local predictor is used with NT as starting state
   (iv) has 90% prediction accuracy when 1-bit local predictor is used with T as starting state

(a) (i) and (iii)
(b) (ii) and (iii)
(c) (i), (ii), (iv)
(d) (ii) and (iv)

Section C [38 marks] (Note: answer this section in a separate sheet)

Q1. [a] In IEEE-754 single precision floating point representation, how may numbers can we represent in the interval [10, 16)? You may leave your answer in powers of 2. **[4]**

[b] If we use 7 exponent bits, a denormalized exponent of -62, and 24 mantissa bits in floating point, what is the largest positive number power of 2 that we can multiply with 1 to get underflow? **[2]**

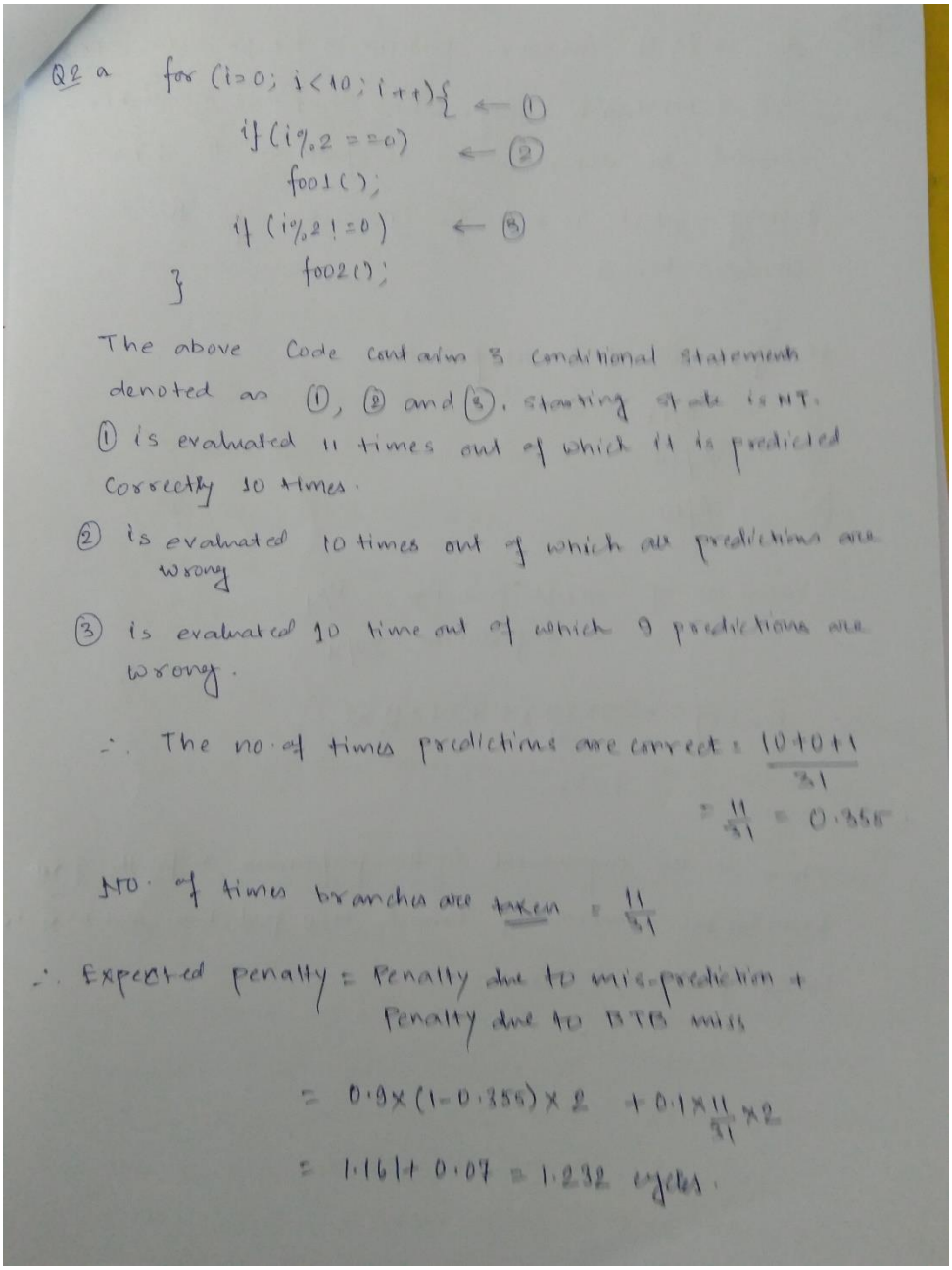


Q2. [a] Consider the following code:

```
for (i=0; i<10; i++){
      if(i%2==0)
            foo1();
      if(i%2!=0)
            foo2();
}
```

Assume that you use a 1-bit local predictor for branch prediction. You use a BTB which has a hit-rate of 90%. Also, a branch miss-prediction (assuming instructions are already in BTB) or BTB miss leads to a penalty of 2 cycles. What is overall branch penalty that is incurred for the above code. (Assume that prediction always starts at NT state) **[6]**

[b] Assume that now you use a (1,1) correlating predictor instead of a 1-bit local predictor? Is there any performance improvement? If yes, by how much; if no, why? **[3]**



Q2 a

```
for (i=0; i<10; i++){   <- (1)
    if (i%2 == 0)       <- (2)
        foo1();
    if (i%2 != 0)       <- (3)
        foo2();
}
```

The above code contains 3 conditional statements denoted as (1), (2) and (3). Starting state is NT.

(1) is evaluated 11 times out of which it is predicted correctly 10 times.

(2) is evaluated 10 times out of which all predictions are wrong

(3) is evaluated 10 time out of which 9 predictions are wrong.

∴ The no. of times predictions are correct = $\frac{10+0+1}{31}$

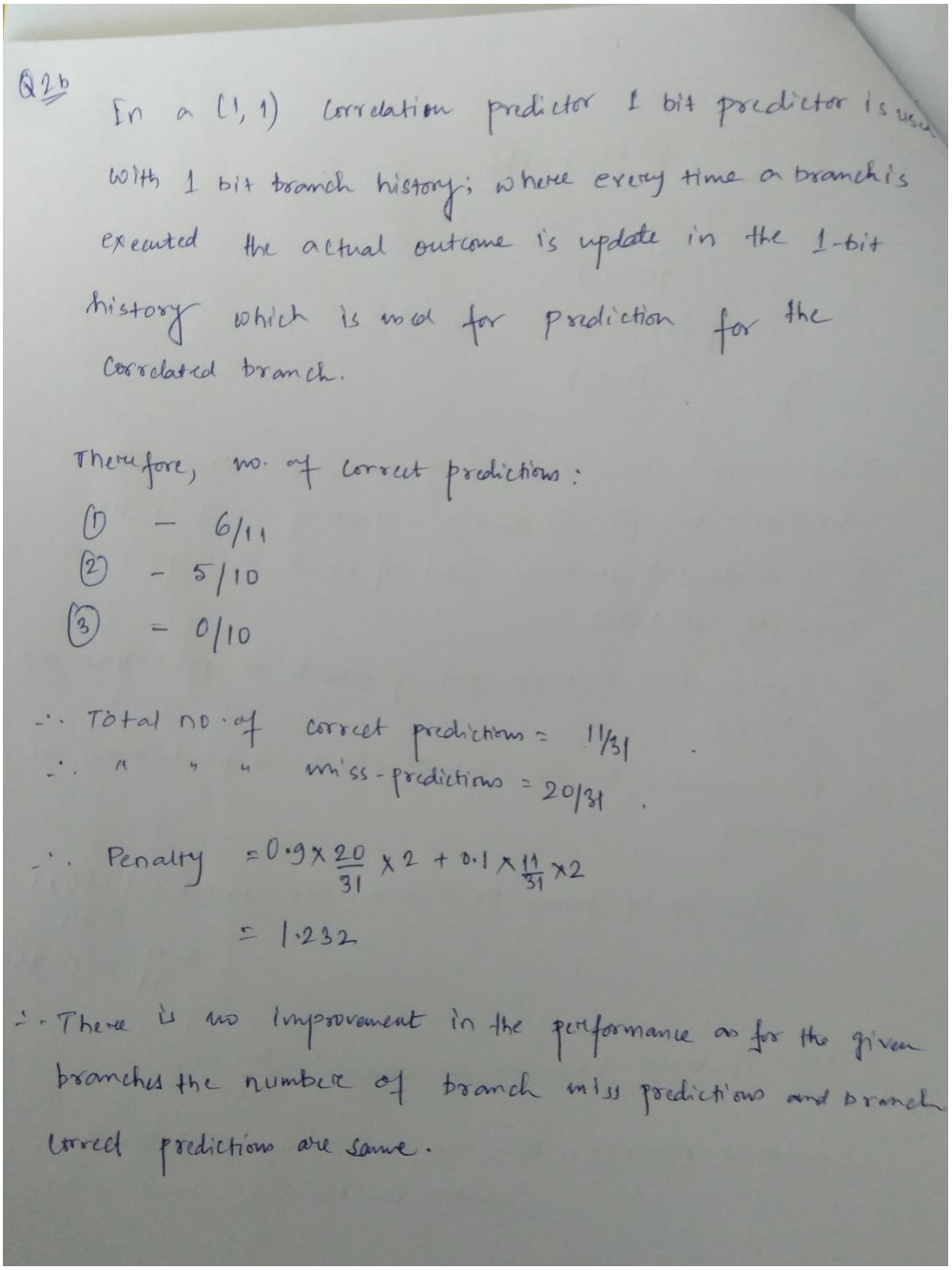
$= \frac{11}{31} = 0.355$

No. of times branches are taken = $\frac{11}{31}$

∴ Expected penalty = Penalty due to mis-prediction + Penalty due to BTB miss

$= 0.9 \times (1-0.355) \times 2 + 0.1 \times \frac{11}{31} \times 2$

$= 1.161 + 0.07 = 1.232$ cycles.



Q3. A program is stored in a 32MB main memory that is attached to a 4KB direct mapped cache with a block size of 16 bytes. The program reads 4 data words viz., A, B, C, D (in that order) 200 times. Let the physical addresses of A, B, C, D be 0x0764420, 0x0764424, 0x176442C, and 0x0764428, respectively. Assume that caches are empty initially and one word = 4 bytes. Find the number of cache hits. Now assume that we use a 2-way set associative cache memory. What is the percentage increase/decrease in cache hit? **[5+3=8]**

Q3 Each data word is read 200 times. Hence total accesses to cache is $200 \times 4 = 800$

address of A = 0x0764420
" " B = 0x0764424
" " C = 0x176442C
" " D = 0x0764428

Given memory size = 32MB = 2^{25} bytes.

∴ Physical address size = 25 bits.

Cache size = 4KB = 2^{12} bytes.

block size = 16 bytes = 2^4 bytes.

word size = 4 bytes = 2^2 bytes.

∴ Number of index bits in the direct mapped cache = $\frac{2^{12}}{2^4} = 2^8 = 256$.

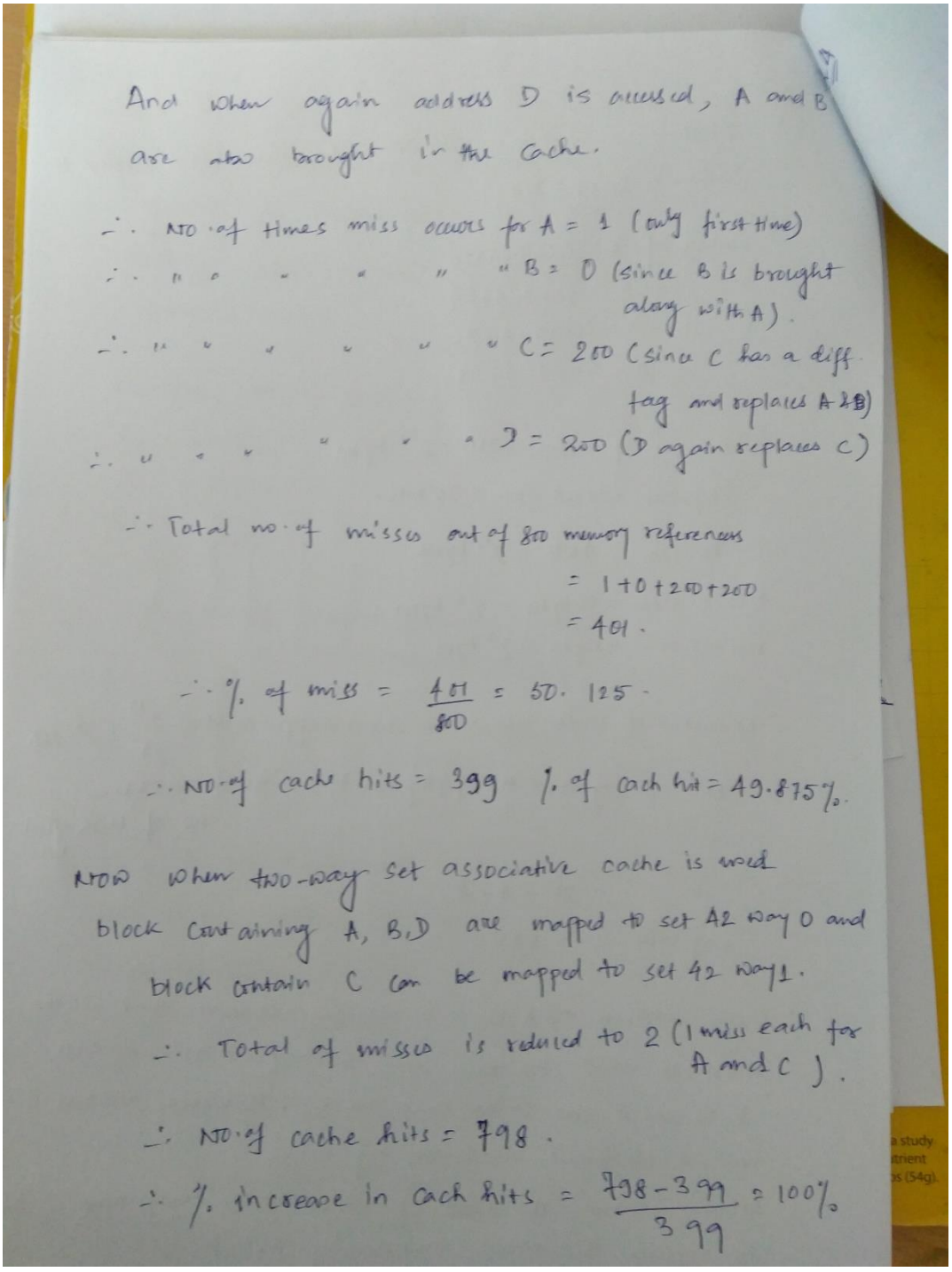
∴ No. of tag bits = No. of physical bits − no. of index bits − no. of block offset bits

$= 25 - 8 - 4$

∴ tag size = 13 bits

∴ From address of A, B, C, D we can observe that all the addresses map to the same set. Moreover, address A, B and D are in the same block. However, address C has a different block and tag bits.

Now, given the order of cache access, every time C has to replace the block containing A, B, D.



Q4. Assume that you have to execute a program/software of size 2GB. The page size is of 16KB. The physical address is of 20 bits and is byte addressable. Assume that the program consists of modules each of size 8KB and only the even numbered modules are accessed in a given run of the program. Calculate the number of page faults that occur, assuming initially TAB is empty. Now if a TAB miss results in a penalty of 100ms with a TAB access time of 20ms and a miss-ratio is 10%, what is effective memory access time. **[4+3=7]**

A and B =

Size of logical address space $= 2^{31}$ bytes

$\therefore$ logical bits = 31.

Size of physical address space $= 2^{20}$ bytes (since it is byte addressable)

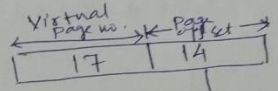
$\therefore$ physical bits = 20.

Given, page size = 16 KB

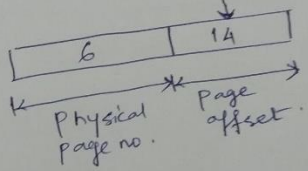
$= 2^{14}$ bytes.

$\therefore$ page offset = 14 bits.

$\therefore$ Virtual address



Physical address



$\therefore$ Size of page table $= 2^{17}$

Now, modules are of size = 8KB. Each page contains two modules (assuming modules are sequentially stored).

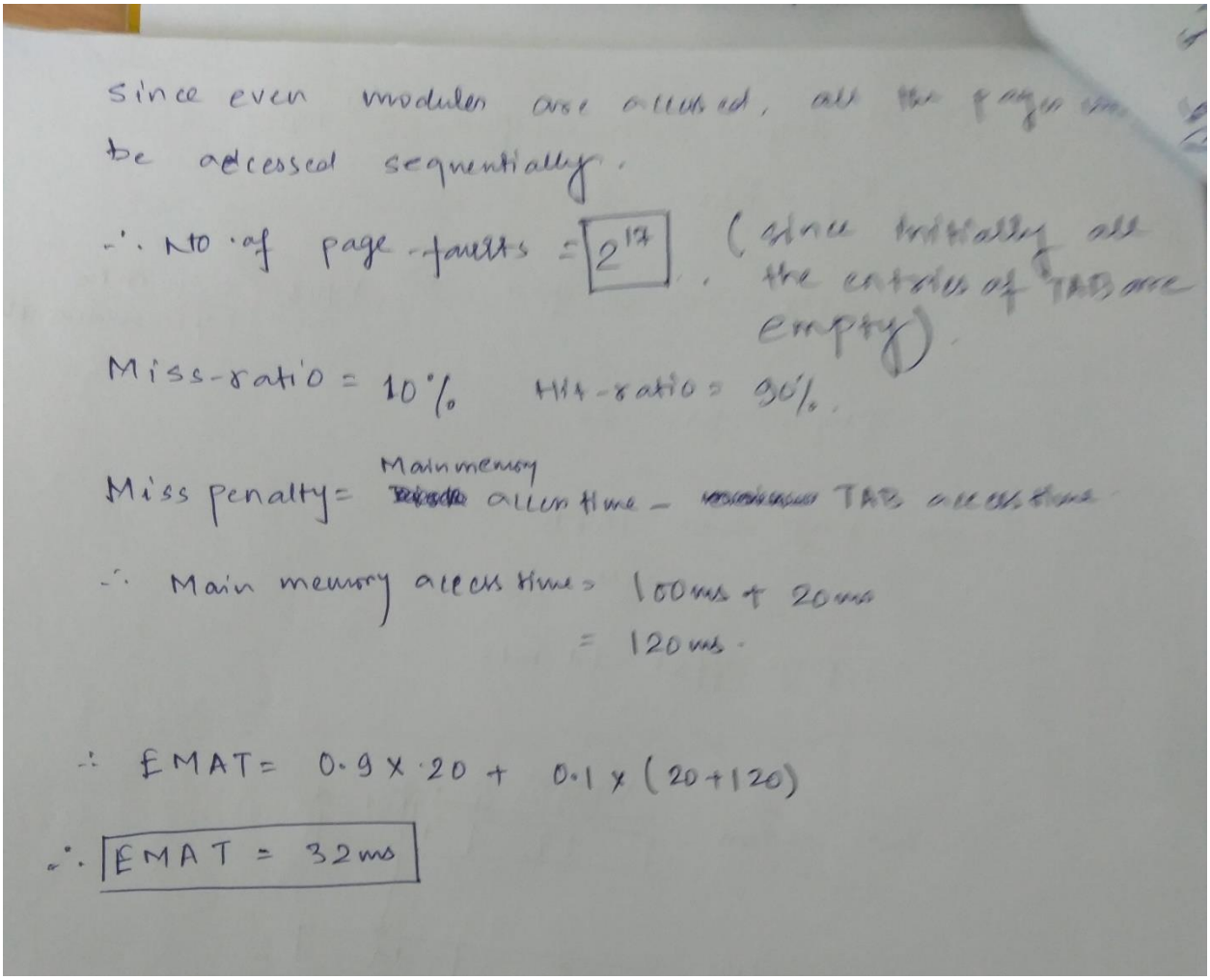
$\therefore$ Module 0, 1 → page 0

Module 2, 3 → page 1

No. of module $= \frac{2^{31}}{2^{13}}$

$= 2^{18}$.

Module $2^{18}-2, 2^{18}-1$ → page $2^{17}-1$



Q5. Consider the following MIPS code:

```
sub $t2, $t1, $t3
slt $t4, $t5, $t4
beq $t4, $zero, target_address
lw $t1, 80($t5)
```

Identify all the data dependencies which the above code has. How many clock cycles will be required to complete the execution of the above code, without any optimization? Now assume that you have a compiler which is capable to performing an optimal schedule by re-arranging the instructions. Is there any improvement in the number of cycles required? Show by drawing a timing diagram. [Assume: there is operand forwarding from EX stage to MEM stage] **[2+3+3=8]**

I_1: Sub $t2, $t1, $t3
I_2: Slt $t4, $t5, $t4
I_3: beq $t4, $zero, target_address
I_4: lw $t1, 80($t5)

RAW: I_2 to I_3 through $t4
WAR: I_1 to I_4 through $t1.
WAW: NIL

	1	2	3	4	5	6	7	8	9
I_1	IF	ID	EX	MEM	WB				
I_2		IF	ID	EX	MEM	WB			
I_3			IF	-	ID	MEM	WB		
I_4					IF	ID	EX	MEM	WB

9 cc required to complete the above instructions.

Instruction re-arranged

I_1: Sub $t2, $t1, $t3
I_2: Slt $t4, $t5, $t4
I_4: lw $t1, 80($t5)
I_3: beq $t4, $zero, target_address.

	1	2	3	4	5	6	7	8
I_1	IF	ID	EX	MEM	WB			
I_2		IF	ID	EX	MEM	WB		
I_4			IF	ID	EX	MEM	WB	
I_3				IF	ID	EX	MEM	WB

No. of cc required = 8

↳ Data is now already available in the $t4 due to operand forwarding from I_2.