CLASS TEST

Sl.: 02 SK_CS_ABCD_090623

MADE EASY
India's Best Institute for IES, GATE & PSUs

Delhi | Bhopal | Hyderabad | Jaipur | Pune | Bhubaneswar | Kolkata

Web: www.madeeasy.in | **E-mail:** info@madeeasy.in | **Ph:** 011-45124612

COMPUTER ORGANIZATION

COMPUTER SCIENCE & IT

Date of Test : 09/06/2023

ANSWER KEY ➤

1.	(d)	7.	(d)	13.	(a)	19.	(b)	25.	(c)
2.	(a)	8.	(c)	14.	(b)	20.	(c)	26.	(b)
3.	(a)	9.	(b)	15.	(a)	21.	(d)	27.	(c)
4.	(c)	10.	(c)	16.	(c)	22.	(b)	28.	(d)
5.	(b)	11.	(a)	17.	(b)	23.	(a)	29.	(a)
6.	(c)	12.	(b)	18.	(b)	24.	(a)	30.	(c)

DETAILED EXPLANATIONS

1. (d)

- Since, vertical microprogram encode the control signals hence a signal decoder is needed which decrease the operation speed of vertical micro-programming in comparison with horizontal micro-programming.
- Since, the control signal bits under horizontal microprogram control unit are not encoded. Hence no signal decoder is needed.

2. (a)

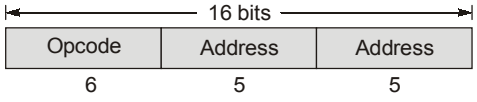
Since, it uses horizontal micro-programmed that requires 1 bit control / signal.
For 125 control signal, we need 125 bits.
Total number of micro-operation instruction = $215 \times 6 = 1290$
It requires 11 bit.

3. (a)

A vectored interrupt is the one, where CPU actually knows the address of the ISR in advance, with the help of an interrupt vector, the interrupting device supplies the branch information to the processor.

4. (c)

Address format



Number of operations = $2^6 = 64$
Number of free opcodes after 2-address = $64 - 2 = 62$
Number of 1 add instruction = $62 \times 32 = 1984$
Free opcodes = $1984 - 1024 = 960$
Number of 0 add instruction = $960 \times 32 = 30720$

5. (b)

- More than one word are put in one cache block to explicit in the spatial locality of reference.
- By the help of virtual memory, programs can exceed from the size of primary memory, hence increases the degree of multi programming.
- Increasing RAM will result in fewer page faults.

Hence only S_2 is the correct statement.

6. (c)

Multiplier	Pair with ($q - 1$)	Recorder
0	0	0
1	0	–1
1	1	0
0	1	+1
1	0	–1
0	1	+1
0	0	0
1	0	–1
0	1	+1
1	0	–1
1	1	0
0	1	+1

7. (d)

RAW hazards: $I_1 - I_2$, $I_1 - I_4$, $I_2 - I_3$, $I_2 - I_4$, $I_2 - I_5$

WAW hazards: $I_2 - I_4$

WAR hazards: $I_3 - I_4$ and $I_2 - I_4$

8. (c)

Rate of transfer to or from any one disk = 30 MBps.

Maximum memory transfer rate = $\frac{4\text{ B}}{10\times10^{-9}} = 400\times10^{6}\text{Bps} = 400\text{ MBps}$

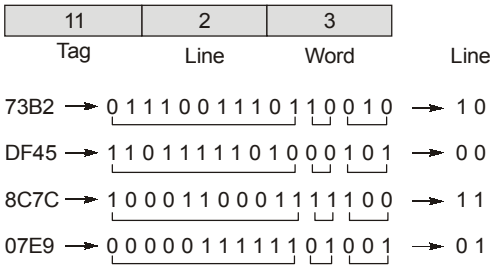
Since rate of data transfer = 30 MBps

Here number of disk transfer = $\left\lceil\frac{400}{30}\right\rceil = 13$

Therefore, 13 disks can simultaneously transfer data to or from the main memory.

9. (b)

The address division will be given as:



10. (c)

- In pipelined CPU, there will be buffer delays. So, for single instruction non-pipelined CPU takes less time compared to pipelined CPU.
- Structural dependencies cause hazards during pipelining.

11. (a)

For non-pipelined processor:

For p-instruction execution time = $\frac{(p\times5)}{5} = p\text{ ns}$

Pipelined processor:

For p-instruction execution time = $\frac{p}{3} = 0.33p\text{ ns}$

Speed-up = $\frac{p}{0.33p} = 3.03$

12. (b)

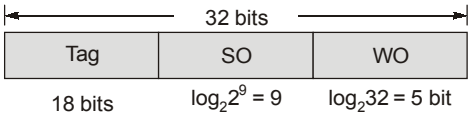
Memory mapped I/O uses the same address bus to address both memory and I/O devices the memory and registers of the I/O devices are mapped to address values.

So, when an address is accessed by the CPU, it may refer to a portion of physical RAM, but it can also refer to memory of I/O device.

13. (a)

$$\text{Number of lines} = \frac{64K}{32} = 2^{11}$$

$$\text{Number of sets} = \frac{2^{11}}{4} = \frac{2^{11}}{2^2} = 2^9$$



$$\begin{aligned}\text{Tag memory size} &= S \times P \times \#\text{ tag bits}\\ &= 2^9 \times 4 \times (18 + 1 + 1 + 2)\\ &= 2^9 \times 2^2 \times 22 \text{ bits}\\ &= 2^{10} \times 2 \times 22 \text{ bits}\\ &= 44 \text{ K bits}\end{aligned}$$

14. (b)

Programmed I/O: Processor issues an I/O command, on behalf of a processor, to an IO module; that process then busy-waits for the operation to be completed before proceeding.

Interrupt driven I/O: The processor isues an IO command on behalf of a process, continues to execute subsequent instruction, and is interrupted by the IO module when the latter has completed its work.

Direct memory access: A DMA module controls the exchange of data between main memory and IO module.

15. (a)

Number of interupts generated = 5000 interrupt / sec

Time by 1 interrupt = 200 μsec

Time consumed by every interrupt = 250 μs

Fraction of processor time consumed = 200/250 = 0.8

In % = 0.8 × 100 = 80

16. (c)

$$\text{Speedup (S)} = \frac{1}{(1-\text{Cache \% used}) + \left[\frac{\text{Cache \% used}}{\text{Speedup using cache}}\right]}$$

$$= \frac{1}{(1-F) + \left(\frac{F}{S}\right)} = \frac{1}{(1-0.9) + \left(\frac{0.9}{30}\right)} = \frac{1}{(0.1) + \left(\frac{0.9}{30}\right)}$$

$$= \frac{30}{3.9} = 7.69$$

17. (b)

	Instruction	Instruction size	Location
I_1	Load r_0, 300	2 word	2000-2003
I_2	MOV r_1, 5000	2 word	2004-2007
I_3	MOV r_2, (r_1)	1 word	2008-2009
I_4	Add r_0, r_2	1 word	2010-2011
I_5	MOV, 6000, r_0	2 word	2012-2015
I_6	HALT	1 word	2016-2017

$\therefore$ Since 1 word is of 2 bytes.

If an interrupt occurs, the CPU has been halted after executing the HALT instruction, the return address 2016 is saved in the stack.

18. (b)

$$\begin{aligned} T_{avg} &= h_1 t_1 + (1 - h_1)h_2 (t_2 + t_1) + (1 - h_1)(1 - h_2)(t_3 + t_2 + t_1) \\ &= 0.65 \times 0.02 + 0.35 \times 0.45 \times 0.22 + 0.35 \times 0.55 \times 2.22 \\ &= 0.013 + 0.03465 + 0.42735 \\ &= 0.475 = 475\ \mu sec \end{aligned}$$

19. (b)

$$\text{Biased exponent} = 18 + 64 = 82$$

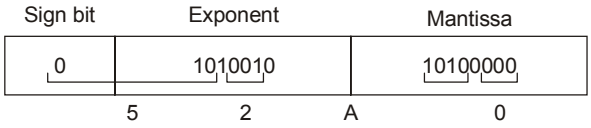
Representing 82 in binary

$$(82)_2 = (1010010)_2$$

Representing mantissa in binary

$$(0.625)_{10} = (0.10100000)$$

Floating point representation is as follows:



20. (c)

Instruction	1	2	3	4	5	6	7	8	9	10	11	12	13
I_1	FI	DI	FO	EI	WO								
I_2		FI	DI	FO	EI	WO							
I_3			FI	DI	FO	EI	WO						
I_4				FI	DI	FO	EI						
I_5					FI	DI	FO						
I_6						FI	FO						
I_7							FI	DI	FO	EI	WO		
I_8								FI	DI	FO	EI	WO	
I_9									FI	DI	FO	EI	WO
Total = 13 * 12 = 156 ns													

21. (d)

Instruction : "Load 1000"

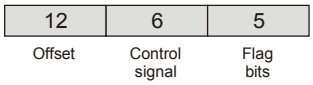
Direct Addressing Mode: Since the content of location '1000' is '1300'. Hence, 1300 will be loaded.

Indirect Addressing Mode: The content of location '1000' is '1300'. The content of memory location '1300' is '1200'. Hence, 1200 will be loaded.

Base (Indexed) Addressing Mode: [1000 + 200] $\Rightarrow$ [1200]. The content of memory location '1200' is 800.

22. (b)

Vertical Microprogramming :



Length of Control Word = Flag + Control Signal + Offset [Offset = $\log_2$ [240 × 12]]

= 5 + 6 + 12

= 23 bits

For 3 Control Words = 23 × 3 = 69 bits.

23. (a)

Assume x % of instructions incur 3 stall cycles.

$$\text{Speedup} = \frac{\text{Pipeline depth}}{(1+ \#\text{stall/ Instruction})}$$

Number of stall / Instructions = (% Instruction don't incur stall × 0 + % Instruction incur stall × 3)

$= (1 - x) \times 0 + x\,(3) = 3x$

$$\therefore \quad 3 = \frac{5}{1+3x}$$

$3 + 9x = 5$

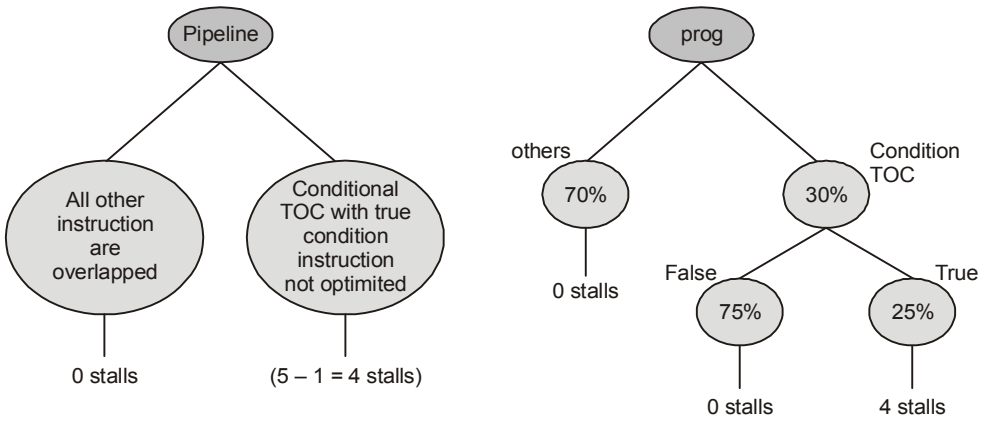
$9x = 5 - 3 = 2$

$x = 2/9 = 0.222$

In % = 0.222 × 100 = 22.2

24. (a)

$$\text{Cycle time} = \frac{1}{\text{Clock time}} = \frac{1}{20\ \text{GHz}} = 0.05\ \text{nsec}$$



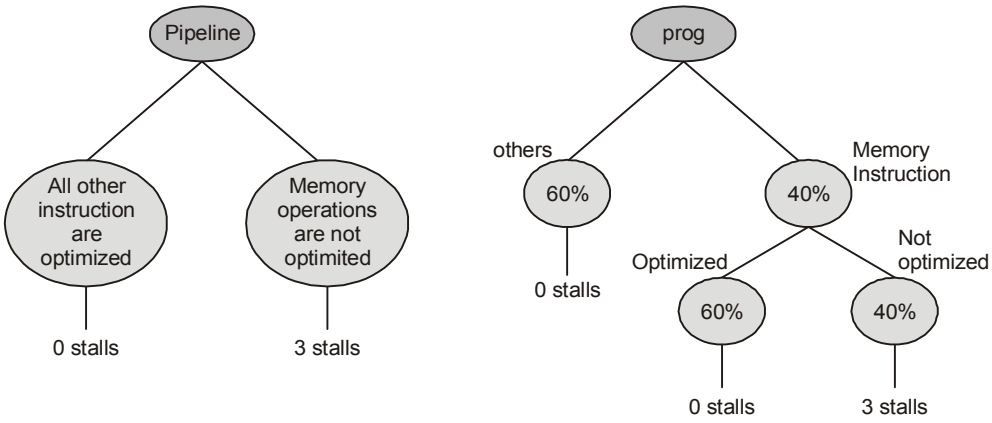
Number of stalls/Instruction = (0.3 × 0.25 × 4) = 0.3

Average instruction ET = (1 + # stalls / Instruction) × Cycle time

= (1 + 0.3) × 0.05 ns

= 0.065 ns

25. (c)



Number of stalls/Instruction = $(0.4 \times 0.4 \times 3) = 0.48$

Average instruction ET = (1 + # stalls / Instruction) × Cycle time = $(1 + 0.48) \times$ 8 ns = 11.84 ns

26. (b)

Stack	
Push (A)	8 + 64
Push (B)	8 + 64
Add	8
Pop (C)	8 + 64
= 224 bits	

Accumulator	
Load A	8 + 64
Add B	8 + 64
Store C	8 + 64
= 216 bits	

Register-Memory	
Load R_1, A	8 + 6 + 64
Add R_3, R_1, B	8 + 6 + 6 + 64
Store R_3, C	8 + 6 + 64
= 240 bits	

Load-store	
Load R_1, A	8 + 6 + 64
Load R_2, B	8 + 6 + 64
Add R_3, R_1, R_2	8 + 6 + 6 + 6
Store R_3, C	8 + 6 + 64
= 260 bits	

Total size = 224 + 216 + 240 + 260 = 940 bits

27. (c)

251 opcodes ⇒ 8 bits for each register

∴

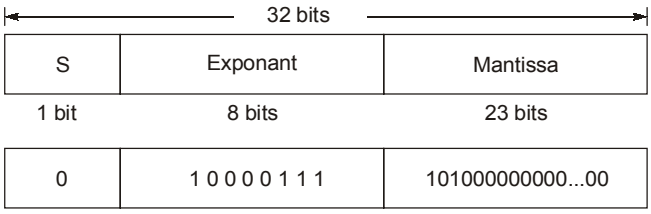
Opcode	R_1, R_2, R_3, R_4

8 bits + 5 × 4 bits = 8 + 20 = 28 bits

[∵ 4 for 4 registers i.e., R_1, R_2, R_3, R_4].

28. (d)

Format of single precision floating point is



Value = $1.M \times 2^{E-127}$

= $1.1010 \times 2^{135-127}$

= $(1.1010)_2 \times 2^8$

= $1.625 \times 2^8 = (416)_{10}$

Octal representation

8	416	
8	52	0
	6	4

Octal representation is (640).

29. (a)

Number of registers in RISC = G + W (L + C)

208 = G + 16 (8 + 4)

G = 16

30. (c)

When instruction is a computation:

Memory reference : Fetch instruction
Fetch reference of the operand
Fetch operand

Total 3 memory references.

When instruction is a branch:

Memory reference : Fetch instruction
Fetch operand reference and loading program counter

Total 2 memory references.

■■■■