

IAT 1

Answer all questions under all the sections

*** Required**

1. Email address *

2. Name *

3. USN *

4. Section *

Mark only one oval.

A

B

E

5. In computers, subtraction is usually carried out by *

2 points

Mark only one oval.

- 1's complement
- 2's complement
- 9's complement
- 10's complement

6. What characteristic of RAM makes it not suitable for permanent storage? *

2 points

Mark only one oval.

- Too slow
- Unreliable
- Non-volatility
- Volatility

7. Computers use addressing modes for *

2 points

Mark only one oval.

- giving programming versatility
- to reduce number of bits in the field of instruction
- specifying rules for modifying address field of the instruction
- all of the above

8. The circuit used to store one bit of data is *

2 points

Mark only one oval.

- register
- encoder
- flip flop
- decoder

9. FFCD9 h is equivalent to *

2 points

Mark only one oval.

- (1111 1111 1100 1101 1001) base 2
- (10809) base 10
- both of the above
- none of these

10. Cache memory acts between *

2 points

Mark only one oval.

- CPU and RAM
- RAM and ROM
- CPU and Hard Disk
- None

11. A 3-input NOR gate gives logic high output only when *

2 points

Mark only one oval.

- all input are low
- one input is high
- two inputs are high
- one input is low

12. A binary digit is called a *

2 points

Mark only one oval.

- byte
- word
- bit
- none

13. The 2's complement of the number 0111 in 6 digit form is *

2 points

Mark only one oval.

- 111001
- 111111
- 110000
- none

14. The load instruction is mostly used to designate a transfer from memory to a process register known as *
- 2 points

Mark only one oval.

- Instruction register
- Program counter
- Accumulator
- Memory address register

15. Consider a hypothetical processor with an instruction of type LW R1 , 20 (R2), which during execution reads a 32-bit word from memory and stores it in a 32-bit register R1. The effective address of the memory location is obtained by the addition of a constant 20 and the contents of register R2. Which of the following best reflects the addressing mode implemented by this instruction for the operand in memory? *
- 2 points

Mark only one oval.

- Immediate Addressing
- Register Addressing
- Register Indirect Scaled Addressing
- Indexed Addressing

16. Which of the following is/are true of the auto-increment addressing mode? I. It is useful in creating self-relocating code II. If it is included in an Instruction Set Architecture, then an additional ALU is required for effective address calculation III. The amount of increment depends on the size of the data item accessed *
- 2 points

Mark only one oval.

- I only
- II only
- III only
- II and III only

17. The value of a float type variable is represented using the single-precision 32-bit floating-point format of the IEEE-754 standard that uses 1 bit for sign, 8 bits for the biased exponent, and 23 bits for mantissa. A float type variable X is assigned the decimal value of -14.25 . The representation of X in hexadecimal notation is *

3 points

Mark only one oval.

- C1640000H
- 416C0000H
- 41640000H
- C16C0000H

18. Scan and upload the working of the previous question.

3 points

Files submitted:

19. Consider the following program segment. Here R1, R2 and R3 are the general purpose registers. Assume that the content of memory location 3000 is 10 and the content of the register R3 is 2000. The content of each of the memory locations from 2000 to 2010 is 100. The program is loaded from the memory location 1000. All the numbers are in decimal. Assume that the memory is word addressable. The number of times ADD instruction is executed is: *

Instruction	Operation	Instruction size (no. of words)
MOV R1, (3000)	$R1 \leftarrow M[3000]$	2
LOOP: MOV R2, (R3)	$R2 \leftarrow M[R3]$	1
ADD R2, R1	$R2 \leftarrow R1 + R2$	1
MOV (R3), R2	$M[R3] \leftarrow R2$	1
INC R3	$R3 \leftarrow R3 + 1$	1
DEC R1	$R1 \leftarrow R1 - 1$	1
BNZ LOOP	Branch on not zero	2
HALT	Stop	1

Mark only one oval.

- 20
- 10
- 11
- 21

This content is neither created nor endorsed by Google.

Google Forms