Quiz 01 Suggestions

1. What do you understand by non-stored program computer?

Ans: A non-stored program computer refers to a computing system where
instructions are executed directly from hardware without being stored in
memory.

2. What do you understand by stored program computer?

Ans: A stored program computer is a type of computer where instructions
are stored in memory along with data.

3. How programs and data are stored in a stored program computer?

Ans: Programs are represented as machine code while data
is stored in memory with it's binary representation.

4. What is the benefit of a stored program computer over a non-stored program computer?

Ans: The benefit of a stored program computer is user can easily modify
and manipulate data without physically rebuilding the hardware,

5. How does the CPU identify instructions from data?

Ans: There is a register in CPU named `instruction pointer` that always point to the next instruction.

6. What does memory hold? What are the contents of memory?

Ans: Memory holds machine codes of instructions and binary representation of data.

7. How many bits are used to represent data?

Ans: The number of bits used to represent data depends on the amount of
information. Common sizes include 8 bits (1 byte), 16 bits, 32 bits, and
64 bits and so on.

8. How many bits are used to represent instruction?

Ans: It depends on the size and complexity of instruction. Common size
for a instruction set is 8 bits, but modern processor usually use 32 or
64 bit for more complex instruction sets.

9. How memory devices are designed?

Ans: Memory devices are designed as an array of storage. Each location can hold fixed number of bits.

10. How memory capacity is defined? Give examples?

Ans: Memory capacity is typically defined by the number of binary digits
(bits) it can store. It's often expressed in terms of bytes (8 bits),
kilobytes (KB), megabytes (MB), gigabytes (GB), terabytes (TB),

11. What are Registers?

Ans: Registers are high speed storage within the CPU.

12. How information is stored in memory?

Ans: Information is stored in memory using binary digits (bits), which are represented electronically as either a high or low voltage state (1 or 0).

13. What do you understand by a byte addressable memory?

Ans: If the context of addressable location is 1 Byte (8 bits), it is called Byte addressable memory

14. Identify byte addressable memory: 1KB, 1K x 16bits, 1k x 32bits.

Ans: 1KB

15. How a 16 bit data is stored in a byte addressable memory. Give example.

Ans: First we will divide 16 bit in 2 string of 8 bits or 1 byte. Then:
In `big-endian` system, we store the most significant byte at the smallest memory address and the least significant byte at the largest. Or in `little-endian` system, we store the least-significant byte at the smallest address and most significant byte at largest address.

16. How a 32 bit data is stored in a byte addressable memory. Give example

Ans: First we will divide 32 bit in 4 string of 8 bits or 1 byte. Then:
In `big-endian` system, we store the most significant byte at the smallest memory address and the least significant byte at the largest. Or in `little-endian` system, we store the least-significant byte at the smallest address and most significant byte at largest address.

17. How a 64 bit data is stored in a byte addressable memory. Give example.

Ans: First we will divide 64 bit in 8 string of 8 bits or 1 byte. Then:
In `big-endian` system, we store the most significant byte at the smallest memory address and the least significant byte at the largest. Or in `little-endian` system, we store the least-significant byte at the smallest address and most significant byte at largest address.

18. What is Big endian system? Give example

Ans: In Big Endian System we store the most significant byte at the smallest memory address and the least significant byte at the largest.

19. What is Little endian system? Give example

Ans: In little endian system we store the least-significant byte at the smallest address and most significant byte at largest address.

20. What do you understand by memory address?

Ans: Memory locations are addressed as some code. They are memory address.

21. How many bits are used to represent memory address?

Ans: 8 bits.

22. How many bits will require to address all addressable locations of a 1KB memory?

Ans: 1KB = 1024 Bytes

=> $2^n = 1024$

=> $2^n = 2^{10}$

=> n = 10

23. How many bits will require to address all addressable locations of a 4KB memory?

Ans: 4KB = 4 x 1024 Bytes

=> $2^n = 4096$

=> $2^n = 2^{12}$

=> n = 12

24. How many bits will require to address all addressable locations of a 1MB memory?

Ans: 1MB = 1024 x 1024 Bytes

=> $2^n = 1024^2$

=> $2^n = 2^{20}$

=> n = 20

25. How many bits will require to address all addressable locations of a 1GB memory?

Ans: 1GB = 1024x 1024 x 1024 Bytes

=> $2^n = 1024^3$

=> $2^n = 2^{30}$

=> n = 30

26. Show the address of first and highest addressable location of a 1KB memory in binary and hexadecimal format.

Ans: 1KB = 1024 Bytes

=> $2^n = 1024$

=> $2^n = 2^{10}$

=> n = 10

total 1024 locations
lowest addressable location in binary: `0000000000`
lowest addressable location in hexa: `0x000`
highest addressable location in binary: `1111111111`
highest addressable location in hexa: `0x3FF`

27. Show the address of first and highest addressable location of a 4KB memory in binary and hexadecimal format .

Ans: 4KB = 4 x 1024 Bytes

=> $2^n = 4096$

=> $2^n = 2^{12}$

=> n = 12

lowest addressable location in binary: `000000000000`
lowest addressable location in hexa: `0x0000`

highest addressable location in binary: `111111111111`
highest addressable location in hexa: `0xFFF`

28. Show the address of first and highest addressable location of a 16KB memory in binary and hexadecimal format.

Ans: 16KB = 16 x 1024 Bytes

=> $2^n = 2^4 . 2^{10}$

=> $2^n = 2^{12}$

=> n = 12

lowest addressable location in binary: `00000000000000`
lowest addressable location in hexa: `0x00000`
highest addressable location in binary: `11111111111111`
highest addressable location in hexa: `0x3FFF`

29. Show the address of first and highest addressable location of a 1MB memory in binary and hexadecimal format.

Ans: 1MB = 1024 x 1024 Bytes

=> $2^n = 1024^2$

=> $2^n = 2^{20}$

=> n = 20

lowest addressable location in binary: `00000000000000000000`
lowest addressable location in hexa: `0x00000`
highest addressable location in binary: `11111111111111`
highest addressable location in hexa: `0xFFFFF`

30. Show the address of first and highest addressable location of a 1GB memory in binary and hexadecimal format.

Ans: 1GB = 1024x 1024 x 1024 Bytes

=> $2^n = 1024^3$

=> $2^n = 2^{30}$

=> n = 30

lowest addressable location in binary: `000000000000000000000000000000`
lowest addressable location in hexa: `0x00000000`
highest addressable location in binary: `111111111111111111111111111111`
highest addressable location in hexa: `0xFFFFFFFF`

31. Show the address of first and highest addressable location of a 4GB memory in binary and hexadecimal format.

Ans: 4GB = 4 x1024x 1024 x 1024 Bytes

=> $2^n = 2^2 . 2^{30}$

=> $2^n = 2^{32}$

=> n = 32

lowest addressable location in binary: `00000000000000000000000000000000`
lowest addressable location in hexa: `0x00000000`
highest addressable location in binary: `11111111111111111111111111111111`
highest addressable location in hexa: `0xFFFFFFFF`

32. How 1's and 0's are stored in memory?

Ans: In computer memory, 1's and 0's are stored using physical elements that can represent two distinct states. For example `Flip-Flop(by turning on-off)`, `Capacitor(by turning on-off)`, `transistor (by voltage level)`.

33. What is a cell? How a cell is designed? What electronic components are used to design a cell?

Ans: A memory cell is the smallest unit of storage in computer memory. It store a single bit.

34. How many bits are stored in a cell?

Ans: one bit.

35. Calculate the number of cells in 1KB memory.

Ans: Number of cells in 1KB memory = 1024 bytes x 8 cells/byte = 8192 cells

36. Calculate the number of cells in 1K x 4bits memory.

Ans: Number of cells = (1 kilobit) / (4 bits/cell) = 1024 bits / 4bits / cell = 256 cells

37. How memory address selects a memory location? Explain with diagram.

Ans: Memory addresses select memory locations through a process known as address decoding.

38. What is address decoder?

Ans: An address decoder takes the binary address input and activates a specific output line corresponding to the desired memory or device.

39. What is the size of address decoder of 1KB memory?

Ans: For 1KB,

1024 Bytes
=> 2^{10} Bytes

So it requires 10 address lines to address each byte in the memory.

40. What is the size of address decoder of 4KB memory?

Ans: For 4KB,

4 x 1024 Bytes
=> 2^{12} Bytes

So it requires 12 address lines to address each byte in the memory.

41. What is the size of address decoder of 1MB memory?

Ans: For 1MB,

1024 x 1024 Bytes
=> 2^{20} Bytes

So it requires 20 address lines to address each byte in the memory.

42. What is the size of address decoder of 8MB memory?

Ans: For 8MB,

8 x 1024 x 1024 Bytes
=> 2^{23} Bytes

So it requires 23 address lines to address each byte in the memory.

43. Show the architecture of memory

Ans:

Addr	Data
0	01010010
1	11010101
2	11001010
...	....
...	....
1022	11010101
1023	00101010

44. What is RISC?

Ans: RISC stands for Reduced Instruction Set Computer. It's a type of computer architecture design that emphasises a smaller set of simple and frequently used instructions, aiming for high performance by executing these instructions very quickly.

45. What is CISC?

Ans: CISC stands for Complex Instruction Set Computer. It's a type of computer architecture design that features a large set of complex and powerful instructions that can perform multiple operations in a single instruction.

46. What is a microprocessor?

Ans: A microprocessor is a programmable integrated circuit that serves as the central processing unit (CPU) of a computer system. It executes instructions stored in memory to perform arithmetic, logic, control, and input/output operations.

47. What do you understand by a 16-bit microprocessor?

Ans: A 16-bit microprocessor is a type of microprocessor where the ALU is capable of handle 2 16 bit operand at a time.

48. What do you understand by a 32-bit microprocessor?

Ans: A 32-bit microprocessor is a type of microprocessor where the ALU is capable of handle 2 32 bit operand at a time.