# UNIT - I

1. What are different types of computers? Mention their applications.
2. What is register transfer language? With suitable examples, explain the representation of instructions in register transfer language and assembly language.
3. What is a stack? Discuss its organization.
4. Explain following terms: 1) Micro-operation 2) Micro-instruction 3) Pseudo instruction.
5. Draw a diagram of 4-bit binary incrementer and explain it briefly.
6. Represent the following conditional control statement(s) by two register transfer statements with control function. If (P = 1) then (R1← R2) else if (Q=1) then (R1←R3)
7. Which information is stored by Program Counter (PC)?
8. Design a digital circuit for 4-bit binary adder.
9. Write a sequence of micro operation(s) for the following memory reference instructions.
   i) ADD ii) BUN
10. Write a program to evaluate the following arithmetic statement X= [A * (B + C) – D] / (E + F - G)
    (i) Using a general register computer with three-address instructions.
    (ii) Using an accumulator type computer with one-address instructions.
    (iii) Using a stack organized computer with zero-address operation instructions.
11. What do you mean by register transfer? Explain in detail. Also discuss three state bus buffer.
12. Write micro operations for ADD R1, R2.
13. What are the types of micro operations?
14. Discuss about Memory Reference Instructions
15. Write the structure of buses used in computer system?
16. Discuss the usage of MAR and MDR in computer organization?
17. Discuss about micro instruction sequence in fetch instruction cycle?
18. List out the typical logical and bit manipulation instructions.
19. What is register-reference instruction?
20. Explain BCD adder with diagram.
21. What are the different phases a basic computer instruction cycle consists? Explain instruction cycle with flowchart.
22. Explain the complete design of simple system to implement RTL code using direct connections, bus and tri-state buffers.
23. Discuss about stack organization of memory. Give its applications.
24. List and explain the functional units of a computer with a neat diagram.
25. How the data can be read from memory? Explain with timing diagrams for memory read and memory write operations.
26. Explain briefly the three categories of computer programming languages.
27. List and explain the various phases in the instruction execution? Draw and explain how a single-bus data path is organized inside a processor when sequences of instructions are executed.
28. Show the block diagram of hardware that implements the following register transfer statements: T2=R2 ← R1 , R1 ← R2

29. Design a 4-bit combinational circuit decrementer using four full-adder circuits
30. Illustrate one-address and zero-address instruction formats, With Examples?
31. Discuss status bit conditions with Diagram?
32. Explain conditional branch instructions?
33. Define program interrupt? Explain external interrupts and internal interrupts.
34. Can a processor be designed without any condition codes? Justify.
35. Explain the four basic types of operations that need to be supported by an instruction set?
36. An 8-bit register contains the binary value 10011100. What is the register value after arithmetic shift right? Starting from the initial number 10011100, determine the register value after an arithmetic shift left, and state whether there is an overflow
37. Give few examples of external interrupts and few examples of internal interrupts. What is the difference between a software interrupt and subroutine call.
38. Using a 4-bit counter with parallel load and a 4-bit adder, draw a block diagram that shows how to implement the following statements:
    x: R1←R1+R2 Add R2 to R1
    x'y: R1←R1+1 Increment R1 where R1 is a counter with parallel load and R2 is a 4-bit register
39. Explain the following with respect to stack organization
    i) Register stack ii) Stack Operations iii) Reverse Polish Notation
40. Explain the mapping from instruction code to micro instruction address. Give the first micro instruction for the 0010, 1011 and 1111.