SPOS Viva Questions & Answers (Short Notes)

1. What is two pass assembler?

A two-pass assembler scans source twice: Pass-1 builds symbol/literal tables; Pass-2 generates final code.

2. What is the significance of symbol table?

Stores labels and their addresses.

3. What is the use of POOLTAB and LITTAB?

POOLTAB groups literal pools; LITTAB stores literals and addresses.

4. How literals are handled in pass I?

Collected in LITTAB and assigned pool indices.

5. What are the tasks done in Pass I?

Build tables, compute addresses, handle directives.

6. How error handling is done in pass I?

Detect syntax and duplicate label errors.

7. Which variant is used in implementation?

Two-pass assembler.

8. Why?

Handles forward references.

9. Which intermediate data structures are designed and implemented in Pass I?

SYMTAB, LITTAB, POOLTAB, OPTAB.

10. What is the format of a machine code generated in Pass II?

Opcode + operand fields.

11. What is forward reference?

Symbol used before defined.

12. How it is resolved by assembler?

Resolved in Pass-2 after definition found.

13. What is loader?

Loads object code into memory for execution.

14. What are the functions of loader?

Loading, relocating, linking, transferring control.

15. Types of loader?

Absolute, relocating, linking, dynamic.

16. What is relocation?

Adjusting code addresses to new memory.

17. What is linking?

Combining modules and resolving symbols.

18. What is linking loader?

Performs linking while loading.

19. What is dynamic linking?

Links at runtime using shared libraries.

20. What is link editor?

Links object files into executable.

21. What is program relocation?

Changing addresses for execution location.

22. What is relocation register?

Holds base for address translation.

23. Different relocation strategies?

Static, dynamic, position-independent.

24. What is swapping?

Moving processes between RAM and disk.

25. What are overlays?

Load only needed code parts to save memory.

26. What is memory management?

Allocates and manages process memory.

27. What is fragmentation?

Memory wastage: internal/external.

28. How fragmentation is handled?

Compaction or paging/segmentation.

29. What is paging?

Fixed-size page/frame mapping.

30. What is segmentation?

Variable-size logical division.

31. What is demand paging?

Load pages only when referenced.

32. What are the goals of scheduling?

Efficiency, fairness, responsiveness.

33. Which scheduling algorithm is best?

Depends: SJF or RR as per use.

34. Which algorithm is best and why?

SJF min avg wait, RR fair for users.

35. Need of allocating blocks to jobs?

Provide memory for execution.

36. What is the time taken by each algorithm?

Varies by complexity and overhead.

37. What is virtual memory?

Illusion of large memory using disk.

38. Which Page replacement algorithm is best?

Optimal (theoretical), LRU practical.

39. What is Belady s Anomaly?

More frames   more faults (FIFO).

40. When is page replacement used?

When page fault and no free frame.

41. What is page fault?

Page not in memory, fetched from disk.

42. What is paging scheme?

Mapping pages to physical frames.

43. What are counting-based page replacement algorithms?

Use frequency counts (LFU).

44. Which algorithm is best and why?

Optimal theoretical, LRU practical.

45. Need of allocating blocks to jobs?

Provide memory, reduce fragmentation.

46. What is the time taken by each algorithm?

Depends on algorithm complexity.