Programming languages have evolved to include features that support parallelism, which is the ability to execute multiple tasks simultaneously.

Some of the key language features for parallelism are:

1. Threads
2. Synchronization
3. Parallel constructs
4. Parallel data structures
5. Futures and Promises
6. Task-based parallelism

1. Threads

Threads are lightweight processes that can execute independently of each other. Programming languages such as Java, C++, and Python provide support for creating and managing threads. This feature allows the developer to write code that can execute multiple tasks simultaneously.

2. Synchronization:

Synchronization is the process of coordinating the execution of threads to avoid data races and ensure data consistency. Programming languages provide various mechanisms for synchronization, such as locks, semaphores, and barriers.

3. Parallel constructs

Some programming languages provide constructs that allow the developer to specify

parallelism directly in the code. For example, OpenMP and MPI provide constructs for parallel loops, parallel regions, and message passing, which can be used to write parallel code.

4. Parallel data structures

Parallel data structures are data structures that are optimized for parallel access. Examples include parallel arrays, hash tables, and trees. Programming languages such as Cilk and UPC provide support for parallel data structures.

5. Futures and Promises

Futures and Promises are constructs that allow the developer to specify asynchronous execution of tasks. Futures represent the result of a computation that may not have completed yet, while Promises represent a commitment to produce a result in the future. Programming languages such as Scala and Haskell provide support for Futures and Promises.

6. Task-based parallelism

Task-based parallelism is a programming model where the developer specifies tasks that can be executed in parallel. The system schedules the tasks dynamically based on the available resources. Programming languages such as TBB and Cilk Plus provide support for task-based parallelism.

Related posts:

1. Structure of Desktop computers
2. Logic Gates
3. Register Organization

4. Bus structure in Computer Organization
5. Addressing modes
6. Register Transfer Language
7. Numerical problem on Direct mapping
8. Registers in Assembly Language Programming
9. Array in Assembly Language Programming
10. Net 31
11. How to start with GNU Simulator 8085
12. Cache Updating Scheme
13. Cache Memory
14. Principle of Cache Memory
15. Cache Mapping
16. Addition and subtraction in fixed point numbers
17. PCI Bus
18. Booths Algorithm
19. Write a short note on design of arithmetic unit ?
20. Write a short note on Array processors ?
21. Write a short note on LRU algorithm ?
22. What is the format of Micro Instruction in Computer Architecture explain ?
23. What is the layout of pipelined instruction in Computer Architecture ?
24. Explain the following interfaces in Detail:PCI Bus, SCSI Bus, USB Bus
25. What is Memory Organization ? Discuss different types of Memory Organization in Computer System.
26. Computer Organization Q and A
27. Write short note on improving cache performance methods in detail ?
28. What is Multiprocessor ? Explain inter process communication in detail ?
29. Briefly explain the concept of pipelining in detail ?

30. Discuss the following in detail: RISC architecture, Vector processing ?
31. Define the instruction format ? Explain I/O System in detail ?
32. Explain the design of arithmetic and logic unit by taking on example ?
33. Explain how addition and subtraction are performed in fixed point number ?
34. Explain different modes of data transfer between the central computer and I/O device ?
35. Differentiate between Serial and parallel data transfer ?
36. Explain signed magnitude, signed l's complement and signed 2's complement representation of numbers. Find the range of numbers in all three representations for 8 bit register.
37. If cache access time is lOOns, main memory access time is 1000 ns and the hit ratio is 0.9. Find the average access time and also define hit ratio.
38. Explain hardwired microprogrammed control unit ? What is address sequencer circuit ?
39. Explain how a stack organized computer executes instructions? What is Stack?
40. Draw and explain the memory hierarchy in a digital computer. What are advantages of cache memory over main memory?
41. What is Associative memory? Explain the concept of address space and memory space in Virtual memory.
42. What is Paging? Explain how paging can be implemented in CPU to access virtual memory.
43. Explain SIMD array processor along with its architectural diagram ?
44. Write short notes on
45. Draw the functional and structural views of a computer system and explain in detail ?
46. Explain general register organization.
47. Compare and contrast DMA and I/O processors ?
48. Define the following: a) Flynn's taxonomy b) Replacement algorithm
49. Explain the various pipeline vector processing methods ?

50. What are different addressing modes? Explain them.
51. Explain any page replacement algorithm with the help of example ?
52. What is mapping? Name all the types of cache mapping and explain anyone in detail.
53. Explain arithmetic pipeline ?
54. Write short notes on, a) SIMD, b) Matrix multiplication c) Instruction format
55. Differentiate: a) Maskable and non-maskable interrupt b) RISC and CISC
56. Computer Organization Previous Years Solved Questions
57. Booths algorithm to muliyiply +5 and -15