1. Which type of parallelism involves breaking down a problem into smaller tasks that can be executed simultaneously?

a) Data parallelism
b) Functional parallelism
c) Task parallelism
d) Load parallelism

Answer: c) Task parallelism
Task parallelism involves breaking down a problem into smaller tasks that can be executed concurrently.

---

2. What is the primary concern of data parallelism?

a) Distributing tasks among processors
b) Ensuring synchronization between threads
c) Dividing data among processors
d) Managing memory allocation

Answer: c) Dividing data among processors
Data parallelism focuses on distributing data across multiple processors for simultaneous processing.

---

3. Which law of parallel scalability states that the maximum speedup is limited by the sequential portion of the algorithm?

a) Amdahl’s Law
b) Gustafson’s Law
c) Moore’s Law
d) Little’s Law

Answer: a) Amdahl’s Law
Amdahl’s Law states that the maximum speedup of a parallel algorithm is limited by the sequential fraction of the algorithm.

---

4. What does Amdahl’s Law help in understanding?

a) Data parallelism
b) Task parallelism
c) Scalability limits in parallel computing
d) Synchronization techniques

Answer: c) Scalability limits in parallel computing
Amdahl’s Law helps in understanding the limits to the speedup that can be achieved by parallelizing a computation.

---

5. Which metric is used to evaluate the efficiency of parallel algorithms by comparing their performance to an idealized version running on an infinite number of processors?

a) Speedup
b) Efficiency
c) Scalability
d) Load imbalance

Answer: b) Efficiency
Efficiency measures how well a parallel algorithm utilizes the available resources compared to an idealized scenario.

---

6. What factor can cause load imbalance in parallel computing?

a) Efficient synchronization
b) Uneven distribution of tasks
c) Homogeneous processors
d) Low memory consumption

Answer: b) Uneven distribution of tasks
Load imbalance occurs when tasks are not distributed evenly among processors, leading to some processors being underutilized while others are overloaded.

---

7. In shared memory parallel programming, what does OpenMP stand for?

a) Open Multi-Processing
b) Open Memory Parallelization
c) Open Message Passing
d) Open Multiprocessing Protocol

Answer: a) Open Multi-Processing
OpenMP stands for Open Multi-Processing, which is a popular API used for shared memory parallel programming.

---

8. Which OpenMP directive is used to specify the scope of variables in parallel regions?

a) #pragma omp parallel
b) #pragma omp for
c) #pragma omp shared
d) #pragma omp private

Answer: d) #pragma omp private
The `private` directive in OpenMP is used to specify variables with private scope in parallel regions.

---

9. Which OpenMP directive is used to distribute loop iterations among threads?

a) #pragma omp master
b) #pragma omp barrier
c) #pragma omp for
d) #pragma omp critical

Answer: c) #pragma omp for
The `for` directive in OpenMP is used for work-sharing among threads by distributing loop iterations.

---

10. What OpenMP directive is used to synchronize threads at a specific point in the code?

a) #pragma omp master
b) #pragma omp barrier
c) #pragma omp atomic
d) #pragma omp single

Answer: b) #pragma omp barrier
The `barrier` directive in OpenMP is used to synchronize threads at a specific point in the code.

---

11. Which OpenMP directive is used to ensure that only one thread executes a certain block of code?

a) #pragma omp master
b) #pragma omp barrier
c) #pragma omp single
d) #pragma omp critical

Answer: c) #pragma omp single
The `single` directive in OpenMP ensures that a block of code is executed by only one thread.

---

12. What OpenMP directive is used to perform a reduction operation on variables across multiple threads?

a) #pragma omp master
b) #pragma omp barrier
c) #pragma omp atomic
d) #pragma omp reduction

Answer: d) #pragma omp reduction
The `reduction` directive in OpenMP is used to perform reduction operations across multiple threads, such as sum or product.

---

13. Which OpenMP scheduling policy assigns equal-sized chunks of iterations to each thread?

a) Static scheduling
b) Dynamic scheduling
c) Guided scheduling
d) Auto scheduling

Answer: a) Static scheduling
Static scheduling assigns equal-sized chunks of loop iterations to each thread at the beginning of the parallel region.

---

14. In OpenMP, what does the `schedule(static, chunk_size)` clause specify?

a) Equal distribution of iterations among threads
b) Dynamic allocation of iterations to threads
c) Guided allocation of iterations to threads
d) Automatic allocation of iterations to threads

Answer: a) Equal distribution of iterations among threads
The `schedule(static, chunk_size)` clause in OpenMP specifies static scheduling with a specified chunk size.

---

15. Which OpenMP scheduling policy adjusts the chunk size based on the number of remaining iterations?

a) Static scheduling
b) Dynamic scheduling
c) Guided scheduling
d) Auto scheduling

Answer: c) Guided scheduling
Guided scheduling in OpenMP adjusts the chunk size based on the number of remaining iterations to balance the workload.

---

16. In OpenMP, what construct is used for specifying independent tasks that can execute concurrently?

a) #pragma omp parallel
b) #pragma omp sections
c) #pragma omp task
d) #pragma omp master

Answer: c) #pragma omp task
The `task` construct in OpenMP is used for specifying independent tasks that can execute

concurrently.

---

17. Which OpenMP directive is used to synchronize a task's execution with its dependencies?

a) #pragma omp parallel
b) #pragma omp taskwait
c) #pragma omp barrier
d) #pragma omp single

Answer: b) #pragma omp taskwait
The `taskwait` directive in OpenMP is used to synchronize a task's execution with its dependencies.

---

18. In OpenMP, what is the purpose of the `private` clause?

a) To declare variables with thread-private storage
b) To synchronize threads within a parallel region
c) To distribute loop iterations among threads
d) To ensure that only one thread executes a certain block of code

Answer: a) To declare variables with thread-private storage
The `private` clause in OpenMP is used to declare variables that should have private storage

for each thread.

---

19. Which OpenMP directive is used to define a critical section of code that only one thread can execute at a time?

a) #pragma omp parallel
b) #pragma omp sections
c) #pragma omp single
d) #pragma omp critical

Answer: d) #pragma omp critical
The `critical` directive in OpenMP

is used to define a critical section of code that only one thread can execute at a time.

---

20. What is the primary advantage of shared memory parallel programming with OpenMP?

a) Scalability across distributed systems
b) Ease of programming
c) High-level abstraction for task parallelism
d) Low-level control over memory management

Answer: b) Ease of programming

One of the primary advantages of shared memory parallel programming with OpenMP is its ease of programming, as it provides high-level constructs for parallelism.

Related posts:

1. Introduction to Information Security
2. Introduction to Information Security MCQ
3. Introduction to Information Security MCQ
4. Symmetric Key Cryptography MCQ
5. Asymmetric Key Cryptography MCQ
6. Authentication & Integrity MCQ
7. E-mail, IP and Web Security MCQ