**Sample Questions**

Computer Engineering

**Subject Name:** High Performance Computing **Semester: VIII**

Multiple Choice Questions

|  |  |
| --- | --- |
| **Choose the correct option for following questions. All the Questions carry equal marks** | |
| 1. | \_\_\_\_\_\_\_ classified the computers on the basis of organization of the constituent elements in the computer. |
| Option A: | Flynn |
| Option B: | Handler |
| Option C: | Shore |
| Option D: | Feng |
|  |  |
| 2. | Two stage instruction pipeline has |
| Option A: | fetch and Execute instruction |
| Option B: | Fetch and Write Instruction |
| Option C: | Fetch and Decode |
| Option D: | Fetch and Memory Excess |
|  |  |
| 3. | In 3-D hypercube network topology the neighbor of node zero are |
| Option A: | node 1 and node 2 and node 4 |
| Option B: | node 2 and node 3 and node 4 |
| Option C: | node 3 and node 1 and node 4 |
| Option D: | node 1 and node 4 and node 3 |
|  |  |
| 4. | The length of the longest path in a task dependency graph is called |
| Option A: | the critical path length |
| Option B: | the critical data length |
| Option C: | the critical bit length |
| Option D: | the critical byte length |
|  |  |
| 5. | \_\_\_\_\_\_\_\_\_\_\_\_suited to problems that are solved using the divide-and-conquer strategy |
| Option A: | exploratory decomposition |
| Option B: | Recursive Decomposition |
| Option C: | speculative decomposition |
| Option D: | data decomposition |
|  |  |
| 6. | Using fewer than the maximum possible number of processing elements to execute a parallel algorithm is called \_\_\_\_\_\_\_\_\_\_\_\_\_\_ a parallel system in terms of the number of processing elements. |
| Option A: | Scaling down |
| Option B: | Scaling up |
| Option C: | Cost optimal |
| Option D: | Non Cost optimal |
|  |  |
| 7. | Which speedup could be achieved according to Amdahl´s law for infinite number of processors if 5% of a program is sequential and the remaining part is ideally parallel? |
| Option A: | Infinite speedup |
| Option B: | 5 |
| Option C: | 50 |
| Option D: | 20 |
|  |  |
| 8. | Parallelism can be used to increase the (parallel) size of the problem is applicable in \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_. |
| Option A: | Amdahl's Law |
| Option B: | Gustafson-Barsis's Law |
| Option C: | Newton's Law |
| Option D: | Pascal's Law |
|  |  |
| 9. | The Prefix Sum Operation can be implemented using the\_\_\_\_\_\_ |
| Option A: | All-to-all broadcast kernel. |
| Option B: | All-to-one broadcast kernel. |
| Option C: | One-to-all broadcast Kernel |
| Option D: | Scatter Kernel |
|  |  |
| 10. | The \_\_\_\_\_\_\_ functions are used to determine the number of processes |
| Option A: | MPI\_Init |
| Option B: | MPI\_Comm\_world |
| Option C: | MPI\_Comm\_size |
| Option D: | MPI\_Comm\_rank |
|  |  |
| 11. | Handler's classification uses the following three pairs of integers to describe a computer: Computer = (p \* p', a \* a', b \* b')  So here what is a meaning of P’ |
| Option A: | Number of PCUs that can be pipelined |
| Option B: | Number of bits that can be pipelined |
| Option C: | Number of segments can be pipelined |
| Option D: | Number of bytes that can be pipelined |
|  |  |
| 12. | Control hazards occurs due to\_\_\_\_\_\_ |
| Option A: | ADD instruction |
| Option B: | MUL instruction |
| Option C: | DIV instruction |
| Option D: | Branch instruction |
|  |  |
| 13. | Messages in Cut through routing are divided into? |
| Option A: | Packets |
| Option B: | Segments |
| Option C: | Flits |
| Option D: | smaller units |
|  |  |
| 14. | We anticipate which pages we are going to browse ahead of time and issue requests for them in advance is known as \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_. |
| Option A: | Prefetching |
| Option B: | Multithreading |
| Option C: | Multitasking |
| Option D: | Latency |
|  |  |
| 15. | The number and size of tasks into which a problem is decomposed determines the\_\_\_\_\_\_\_\_\_ of the decomposition. |
| Option A: | Concurrency |
| Option B: | Task dependency |
| Option C: | Granularity |
| Option D: | Efficiency |
|  |  |
| 16. | \_\_\_\_\_\_\_\_\_\_\_\_ is due to load imbalance, synchronization, or serial components as parts of overheads in parallel programs. |
| Option A: | Inter process interaction |
| Option B: | Synchronization |
| Option C: | Idling |
| Option D: | Excess computation |
|  |  |
| 17. | Which speedup could be achieved according to Amdahl´s law for infinite number of processors if 5% of a program is sequential and the remaining part is ideally parallel? |
| Option A: | Infinite speedup |
| Option B: | 5 |
| Option C: | 50 |
| Option D: | 20 |
|  |  |
| 18. | Parallelism can be used to increase the (parallel) size of the problem is applicable in \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_. |
| Option A: | Amdahl's Law |
| Option B: | Gustafson-Barsis's Law |
| Option C: | Newton's Law |
| Option D: | Pascal's Law |
|  |  |
| 19. | Synchronization is one of the common issues in parallel programming. The issues related to synchronization include the followings, EXCEPT: |
| Option A: | Deadlock |
| Option B: | Livelock |
| Option C: | Fairness |
| Option D: | Correctness |
|  |  |
| 20. | Which MPI function is used to determine the label of calling process? |
| Option A: | MPI\_Init |
| Option B: | MPI\_Comm\_world |
| Option C: | MPI\_Comm\_size |
| Option D: | MPI\_Comm\_rank |
|  |  |
| 21. | Due to architectural arrangement of a single instruction stream with multiple data streams , array processors machines are called \_\_\_\_\_\_\_\_ array processor. |
| Option A: | MISD |
| Option B: | SIMD |
| Option C: | SISD |
| Option D: | MIMD |
|  |  |
| 22. | SIMD computers require less memory because only \_\_\_\_\_\_\_\_\_\_ needs to be stored. |
| Option A: | one copy of the program |
| Option B: | one instruction of the program |
| Option C: | two instruction of the program |
| Option D: | few instruction of the program |
|  |  |
| 23. | A processor without structural Hazards is \_\_\_\_\_\_\_\_\_\_\_\_. |
| Option A: | Faster |
| Option B: | Stock |
| Option C: | Deadlock |
| Option D: | Structural hazard |
|  |  |
| 24. | Control hazards occurs due to\_\_\_\_\_\_ |
| Option A: | ADD instruction |
| Option B: | MUL instruction |
| Option C: | DIV instruction |
| Option D: | Branch instruction |
|  |  |
| 25. | Pipeline increases the CPU instruction\_\_\_\_\_\_\_\_\_\_\_\_\_. |
| Option A: | Size |
| Option B: | Through put |
| Option C: | Cycle rate |
| Option D: | Time |
|  |  |
| 26. | If during a cycle, no functional units are utilized, this is referred to as \_\_\_\_\_\_ waste |
| Option A: | Horizontal waste |
| Option B: | Vertical waste |
| Option C: | Data waste |
| Option D: | Explicitly waste |
|  |  |
| 27. | If the second instruction cannot be issued because it has a data dependency with the first, only one instruction is issued in the cycle. This is called \_\_\_\_\_\_\_\_\_ issue. |
| Option A: | In-order |
| Option B: | Out-order |
| Option C: | Execution |
| Option D: | Data |
|  |  |
| 28. | Since it uses the out of order mode of execution, the results are stored in \_\_\_\_\_\_ |
| Option A: | Buffers |
| Option B: | Special memory locations |
| Option C: | Temporary registers |
| Option D: | TLB |
|  |  |
| 29. | If an exception is raised and the succeeding instructions are executed completely, then the processor is said to have \_\_\_\_\_\_ |
| Option A: | Exception handling |
| Option B: | Imprecise exceptions |
| Option C: | Error correction |
| Option D: | Exception |
|  |  |
| 30. | The pattern of\_\_\_\_\_\_\_\_\_\_\_ among tasks is captured by what is known as a task-interaction graph |
| Option A: | Interaction |
| Option B: | Communication |
| Option C: | Optimization |
| Option D: | Flow |
|  |  |
| 31. | \_\_\_\_\_\_\_\_\_ mapping techniques distribute the work among processes during the execution of the algorithm. |
| Option A: | Static |
| Option B: | Sequential |
| Option C: | Uniform |
| Option D: | Dynamic |
|  |  |
| 32. | \_\_\_\_\_\_\_\_\_ is a method for inducing concurrency in problems that can be solved using the divide-and-conquer strategy. |
| Option A: | exploratory decomposition |
| Option B: | speculative decomposition |
| Option C: | data-decomposition |
| Option D: | Recursive decomposition |
|  |  |
| 33. | A decomposition into a large number of small tasks is called \_\_\_\_\_\_\_\_\_. |
| Option A: | coarse-grained |
| Option B: | coarse-ungrained |
| Option C: | fine-grained |
| Option D: | fine-ungrained |
|  |  |
| 34. | The number of processors used to execute a program is defined as the \_\_\_\_\_\_\_\_\_\_\_ of parallelism. |
| Option A: | Degree |
| Option B: | Level |
| Option C: | Amount |
| Option D: | Rank |
|  |  |
| 35. | Speed up is defined as a ratio of |
| Option A: | S=Ts/Tp |
| Option B: | S= Tp/Ts |
| Option C: | Ts=S/Tp |
| Option D: | Tp=S /Ts |
|  |  |
| 36. | Total cost of a parallel algorithm is the product of |
| Option A: | Total Cost = Time complexity × Number of processors used |
| Option B: | Total Cost = Time complexity × Number of cycle used |
| Option C: | Total Cost = Time complexity × Number of task used |
| Option D: | Total Cost = Time complexity × Number of instructions used |
|  |  |
| 37. | Most commonly used topologies in message-passing programs are one-, two-, or higher-dimensional grids, that are also referred to as \_\_\_\_\_\_\_\_\_\_ |
| Option A: | Higher Dimensional topology |
| Option B: | Cartesian topologies |
| Option C: | Cart topologies |
| Option D: | Ring topologies |
|  |  |
| 38. | If the parallel program is highly synchronous (i.e., sends and receives are posted around the same time), \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_. |
| Option A: | buffered Receive may perform better than buffered sends |
| Option B: | buffered Receive may perform better than non buffered sends |
| Option C: | buffered sends may perform better than non buffered sends |
| Option D: | non-buffered sends may perform better than buffered sends |
|  |  |
| 39. | The one-to-all broadcast operation is performed in MPI using the \_\_\_\_\_\_\_ |
| Option A: | MPI\_Bcast function |
| Option B: | MPI\_Broadcast function |
| Option C: | MPI\_BroadCast function |
| Option D: | MPI\_BCast function |
|  |  |
| 40. | Non-blocking Message Passing Operations are generally accompanied by a\_\_\_\_\_\_\_\_\_ operation. |
| Option A: | Send Buffer |
| Option B: | Buffer |
| Option C: | check-status |
| Option D: | Receive Buffer |

**Descriptive Questions**

|  |
| --- |
| Explain Decomposition techniques. |
| Write MPI program for Cannon’s Matrix-Matrix Multiplication. |
| Explain different performance metrics for Parallel System. |
| Explain Non-Blocking Communication using MPI. |
| Explain sources of overhead in parallel programs |
| Describe pipeline performance (Efficiency , Speedup and Throughput) w.r.t length of the pipe (n) and task run on pipe(m) for condition m>>n, n>>m and m=n. |
| **Write a MPI program to find sum of N numbers.** |
| Explain speedup, efficiency and scalability with suitable example. |
| **Short note on 'SIMD matrix multiplication'.** |
| State and explain Amdahl’s law. What is the relevance of Amdahl’s law in HPC? |
| Discuss different levels of parallel processing? |
| **With neat block diagram explain in detail about the various programmatic levels of parallel processing.** |
| Explain the different mapping techniques that are used load balancing. |
| Discuss in detail Pipeline hazards with its types. |
| Explain Very long instruction word (VLIW) in detail. |
| Write a parallel MPI program to broadcast a data from root process to 4 other processes. |
| State and Explain the performance metric speed up , Efficiency , Throughput and Scalability |
| Explain in brief classification of parallel system based on memory access. |
| **Discuss the categories of computers based on Handler’s classification.** |
| Explain write-Invalidate Protocol with the help of diagram. |
| Explain Granularity, Concurrency and dependency graph. |
| Write MPI program for broadcast of data. |
| Explain the pros and Cons of Open MP. |
| Explain the Concept of Scatter and Gather. |
| **Explain Quantum Computers.** |
| **Write a short note on Memory organization** |
| **Give the advantages in using non-uniform memory access model.** |
| Explain the pros and Cons of Open MP. |
| Distinguish between loosely coupled and tightly coupled multiprocessors. |
| **Discuss the categories of computers based on Flynn's classification.** |