Final Examination Questions for the 4th-Year Course "Parallel Programming"

1. What are the main characteristics of a distributed system, its architectural principles, and how does it appear to users as a single unified system?
2. Explain parallel computing systems, and describe their role and importance in modern computing systems using practical examples.
3. The main types of parallel data processing are pipeline and parallel. Compare calculations and explain in detail their working mechanism, advantages, and areas of application.
4. According to Flynn's classification, what are the groups into which computing system architectures are divided? Explain their differences with examples .
5. The main technical and architectural characteristics of supercomputers, the main factors ensuring their high performance what with expressed? Examples based on illuminating give .
6. Grid calculation systems concept and its difference from distributed systems and examples of real-life applications based on illuminating give.
7. One of the problems that arise in parallel programming is the data race. Explain the causes and methods of prevention.
8. CUDA technology implement parallel computing on GPU hardware and what is the mechanism for interacting with the CPU ?
9. Compare pipelined and superscalar processing methods and explain the concept of instruction-level parallelism in relation to these technologies.
10. Multi-core Compare multicore and multithreading technologies , their operating principles and application in modern processors examples with illuminating give .
11. Compare SMP (Symmetric Multiprocessing) and AMP (Asymmetric Multiprocessing) systems, and discuss their architectural differences and practical applications . practical examples with critical illuminating give .
12. Describe UMA and NUMA memory architectures, their importance, and their role in modern systems with analytical data.
13. Explain in detail the GPU architecture and its memory hierarchy, the functions of registers, and its importance in parallel programming, based on practical examples.
14. Explain and provide examples of kernel function initialization, memory allocation, and data transfer processes (cudaMalloc, cudaMemcpy) in the CUDA programming model.
15. Cache coherence and its protocols (MESI), Snooping and directory-based approaches are implemented. examples based on comparison , statement arrived give .
16. Explain the concepts of memory management (MMU, page table, page fault) and their role in the operating system. how that statement arrived give .
17. Analyze synchronization mechanisms (mutex, semaphore, barrier, spinlock) in parallel programming . statement eat and their application illuminating give .
18. What is the architecture of heterogeneous computing systems and their role in digital processing reflected in modern artificial intelligence and big data systems?
19. The "3V" model (Volume, Velocity, Variety) in the concept of Big Data and its connection to parallel computing in practice examples based on illuminating give .
20. The main functions and message exchange mechanisms of the MPI (Message Passing Interface) standard in parallel programming in practice in what is expressed ?

21. Explain in more detail the role and main capabilities of OpenMP technology in organizing multithreaded programming in shared memory systems, using examples.
22. OpenMP The working principle of the SPMD (Single Program Multiple Data) model used in the technology and its advantages analytical examples using statement arrived give .
23. Explain the essence of CUDA technology and the parallel computing mechanism based on the interaction between the CPU and GPU using examples.
24. Explain with analytical data the role of pipeline processing in increasing processor performance by dividing instructions into stages.
25. Describe the mechanisms of superscalar architecture that allow multiple instructions to be executed in parallel during a single clock cycle. arrived give .
26. Relate the concept of instruction-level parallelism to pipeline and superscalar technologies , with examples Explain with .
27. Explain the concept of data-level parallelism and how it is used in vector processors and GPUs. give .
28. Compare the concepts of data parallel processing and task parallel processing and identify their main differences. analytical statement eat .
29. Analyze the belonging of the Intel x86 processor to the CISC architecture and the main features of this architecture . examples with Explain.
30. Explain the concept of multi-core computing architecture and its importance in modern computing systems with examples.
31. The working principle of multithreading technology and its role in improving processor efficiency how indicators based on is expressed ?
32. The essence of Intel's Hyper-Threading technology and its mechanism of operation at the hardware level examples based on statement arrived give .
33. The working principle of AMD's " Simultaneous Multithreading (SMT) " technology and its differences from other multithreading technologies . examples using statement eat .
34. Compare the concepts of process and thread in an operating system and identify their main differences. analytical illuminating give .
35. The essence, operating principle, and advantages of the symmetric multiprocessor (SMP) architecture are explained. analytical examples using illuminating give .
36. Explain the essence of the architecture of asymmetric multiprocessor systems (AMP), the control mechanism, and the tasks between processors using analytical examples.
37. Explain the principle of operation of shared memory computing systems and their importance in multiprocessor architectures using analytical examples.
38. The main features of the NUMA (Non-Uniform Memory Access) architecture and the dependence of memory access time on distance analytical examples using illuminating give .
39. Describe the hybrid model of parallel programming and analytically explain the mechanism of joint use of distributed and shared memory approaches in it.
40. Compare CPU and GPU architectures in terms of number of cores and computational capabilities , and illustrate with analytical examples.
41. Examples of the roles and relationships of the concepts of "Host" and "Device" in the CUDA programming model using illuminating give .

42. Describe the structure of a Streaming Multiprocessor (SM) and its functions in GPU computing . analytical examples using illuminating give .
43. List the main levels of the GPU memory hierarchy and illustrate their differences in terms of tasks and processing speed with examples.
44. Graphic in processors (GPU) general Explain the advantages of using shared memory and its impact on parallel computing performance . analytical examples using illuminating give .
45. Explain the context switching problem that occurs when an excessive number of threads are introduced in parallel programs and its impact on system speed using analytical examples.
46. Explain the essence of the Cache Coherence principle and its role in ensuring data consistency in parallel systems with examples. using illuminating explaining give .
47. The principle of operation of the snooping-based coherence mechanism and the process of monitoring data over a common system bus are explained using analytical examples. Explain.
48. Explain the problem of speed mismatch between the processor and main memory and its impact on system performance using analytical examples.
49. The role of the MMU (Memory Management Unit) in translating virtual addresses into physical addresses is discussed in more detail. statement eat .
50. a page fault in parallel programming and how it is handled by the operating system.
51. Examples of the function of the Page Table in a virtual memory system and its role in address mapping using illuminating give . .
52. The main types of computing cluster nodes (compute node and head/login node) and their functions are analyzed. examples using illuminating give .
53. The main purpose of the MPI (Massage Passing Interface) standard for organizing inter-process communication in distributed systems is to provide practical examples using illuminating give .
54. Analyze the basic operations (send/receive) of message exchange between processes in MPI (Massage Passing Interface) technology.
55. Examples of the role of the concept of Memory Consistency in ensuring a consistent view of data in parallel systems based on Explain.
56. Explain the main purpose of thread synchronization in multithreaded programming and its importance in using shared resources.
57. Lock/Mutex synchronization mechanism work principle and each other mutual exclusion in provision role analytical illuminating give.
58. Semaphore and mutex mechanisms comparing them work principles and in use differences analytical examples using illuminating give.
59. The barrier mechanism in parallel programming task and currents in synchronization role examples using illuminating give .
60. " Spinlock" mechanism of use effective circumstances and his/her context exchange with related advantages examples based on Explain.
61. Fine-grained synchronization approach the essence and his/her parallelism level in progress role analytical illuminating give .

62. “ Lock-free” programming concept and then " atomic " synchronization operations based on how organization to be completed Explain .
63. CAS (Compare-And-Swap) " atomic " operation work mechanism and its parallel programming importance analytical examples using illuminating give .
64. In lock-free programming to the surface the upcoming "ABA" problem the essence and to its "Compare-And-Swap" operation the impact Explain .
65. " atomic " operations of the processor information consistency provide and synchronization problems eliminate in the making role analysis as illuminating give .
66. Heterogeneous parallelism the concept and various calculation devices together application through system efficiency increase mechanism examples using in detail Explain.
67. Modern heterogeneous calculation systems to the composition incoming main components and their functions explain , explain arrived give .
68. Like ASIC and TPU in parallel programming special architectures purpose and their general to processors relatively advantages examples using illuminating give .
69. CPU and GPU based heterogeneous in architecture central processor management and calculation in the tasks The role of CUDA technology based on statement arrived give .
70. Graphic processing in giving array -parallel programming necessity the reasons and his/her calculation to the process the impact examples Explain using .
71. Current parallel programming in Big Data issues today opportunities and its data again at work importance statement arrived give .
72. Big Data systems parallelism provide for applicable cluster architecture and MapReduce and Spark platforms role practical examples using illuminating give .
73. Digital to signals processing parallel programming in DSP application sectors and his/her to efficiency the impact practical examples using illuminating give.
74. Graphs with parallel computing in operation done increasing frameworks (NetworkX , GraphX , CUDA Graphs) . tasks and application practical examples using illuminating give.
75. Artificial intellect and deep education parallel architectures in systems of need there is the reasons and their calculation to the efficiency the impact examples using illuminating give.
76. Parallel pit in education applicable main libraries (Horovod , TensorFlow , PyTorch) functional opportunities practical examples using illuminating give .
77. Chorovod library in a distributed GPU environment neuron networks in school role mainly what invisible Will it be? Practical examples using illuminating give .
78. DeepSpeed technology big voluminous models education and the "inference" process in optimization importance practical examples using illuminating give .

79. Machine-made The use of the “Data Parallelism” approach in teaching work principle and gradients synchronization mechanism examples using illuminating give.
80. Neuron networks parameters per GPU memory from the bottom of the "Tensor Parallel" approach used the essence practical examples using illuminating give.
81. Deep the need for GPU/TPU-based parallel computing in educational systems analytical examples using illuminating give
82. Python programming of the language Practical application of the capabilities of the TensorFlow, PyTorch, and Horovod libraries in organizing parallel computing examples using illuminating give .
83. The differences between "Data Parallelism" and "Model Parallelism" explain practical examples using illuminating give .
84. Analyze the working principle of the “Data Parallelism ” approach, i.e. the processes of data partitioning and gradient synchronization. statement examples based on bring .
85. The role of GPU and TPU-based parallel computing architectures in improving performance in deep learning systems practical examples using illuminating give .
86. The application of parallel computing in digital signal processing (DSP) and its impact on performance in real-time systems practical examples using illuminating give .
87. The main reasons for the need for parallel architectures in artificial intelligence systems from what What is it ? What do you think ? practical examples using illuminating give .
88. Explain the application of parallel programming in scientific fields such as bioinformatics, physics, and chemistry and its role in accelerating complex computations using practical examples.
89. Analyzing the parallel computing capabilities of the TensorFlow and PyTorch libraries and their differences examples using illuminating give .
90. Practical application of load balancing mechanisms on performance in parallel computing systems examples using illuminating give .
91. Practical application of Amdahl's law to parallel programming limitations and its impact on system speed examples using Explain.
92. Analyticaly compare and explain the principles of operation and areas of application of SIMD and MIMD architectures in parallel programming.
93. Scientifically explain the differences between synchronous and asynchronous execution models in parallel programming , and practically examples using illuminating give .
94. Graphic in processors Analyze the essence of the “ thread divergence ” problem in digital computing and its impact on performance statement arrived give .
95. Practical criteria for evaluating the effectiveness of parallel algorithms (speedup, efficiency) examples using illuminating give .
96. Practical explanation of the causes of deadlock in parallel programming and methods for preventing it . examples using illuminating give .
97. Analyze the causes of the " race condition " problem in parallel programming and methods for preventing it , and provide practical solutions . examples using illuminating give .

98. " pipeline parallelism " in parallel programming and its impact on performance in practice examples using illuminating give .
99. Analyze the impact of memory hierarchy on performance in GPU architecture and illustrate with examples.
100. Analyze the impact of memory hierarchy on performance in GPU architecture and illustrate it using practical examples.

Question 14: What is the main function of the MPI (Message Passing Interface) protocol? **Answer:** It is a language-independent protocol for programming parallel processes, and its main mechanism is to transmit and receive messages.

Question 15: What is OpenMP? **Answer:** It is an open standard and API that supports programming multithreaded applications on multiprocessor systems with shared memory.

Question 16: What parallel programming model does OpenMP support? **Answer:** SPMD (Single Program Multiple Data) model, in which a single, identical code is used for all parallel threads.

Question 17: What prefix do all functions in OpenMP start with? **Answer:** All functions start with the prefix **omp_** and are written in lowercase.

Question 18: Which function is used to return the astronomical execution time of a program in OpenMP? **Answer: The omp_get_wtime()** function.

Question 19: What is CUDA? **Answer:** It is a parallel computing hardware-software architecture developed by NVIDIA for use with graphics processing units (GPUs).

Question 20: What is meant by pipeline processing? **Answer:** A method of increasing efficiency by dividing the instruction execution cycle into stages for simultaneous execution on separate hardware blocks.

Question 21: What is superscalar processing? **Answer:** A processor architecture with multiple independent execution units (ALUs) that allows multiple instructions to be executed in parallel in a single clock cycle.

Question 22: What is meant by instruction-level parallelism (ILP)? **Answer:** The ability of a processor to execute independent instructions simultaneously using pipelines or superscalar architecture technologies.

Question 23: What is data-level parallelism? **Answer:** The ability to perform the same operation on large amounts of data simultaneously, commonly used in vector and graphics processors.

Question 73: What commands control memory in the Release Consistency model? **Answer:** Synchronization is controlled only by the Lock and Unlock operations of the resource.