

Итоговые контрольные вопросы по дисциплине «ПАРАЛЛЕЛЬНОЕ ПРОГРАММИРОВАНИЕ» (4 курс)

1. Какие основные характеристики и архитектурные принципы распределенных систем, и каким образом они представляются пользователю как единая интегрированная система?
2. Какие роль и значение систем параллельных вычислений (parallel computing) в современной вычислительной технике?
3. Сравните основные виды параллельной обработки данных — конвейерность и параллельность: подробно объясните механизмы их работы, преимущества и области применения.
4. На какие классы подразделяется архитектура вычислительных систем согласно классификации Флинна, и в чем заключаются их отличительные особенности (приведите примеры)?
5. Какие основные технические и архитектурные характеристики суперкомпьютеров, и какие ключевые факторы обеспечивают их сверхвысокую производительность?
6. Объясните концепцию Grid-систем (Grid computing), укажите их отличия от традиционных распределенных систем и приведите примеры практического применения в реальной жизни.
7. Объясните причины возникновения проблемы "гонки данных" (data race) при параллельном программировании и опишите методы ее предотвращения.
8. Каким образом реализуются параллельные вычисления на базе GPU с использованием технологии CUDA, и каков механизм ее взаимодействия с центральным процессором (CPU)?
9. Сравните методы конвейерной и суперскалярной обработки, объяснив концепцию параллелизма на уровне команд (ILP) в контексте данных технологий.
10. Сравните многоядерную архитектуру (multicore) и технологию многопоточности (multithreading), объясните принципы их работы и применение в современных процессорах.
11. Сравните системы симметричной (SMP) и асимметричной (AMP) многопроцессорности: опишите их архитектурные различия и сферы практического применения.
12. Дайте характеристику архитектурам памяти UMA и NUMA, объясните их значение и роль в современных вычислительных системах.
13. Опишите архитектуру графического процессора (GPU), его иерархию памяти, функции регистров и их значение в параллельном программировании.
14. Объясните процессы запуска функций-ядер (kernel), выделения памяти и передачи данных (с использованием функций cudaMalloc, cudaMemcpy) в модели программирования CUDA.
15. Сравните принципы когерентности кэша (cache coherence) и ее протоколы (на примере MESI), а также подходы на основе отслеживания (snooping-based) и на основе каталогов (directory-based).
16. Объясните концепции управления памятью (устройство управления памятью MMU, таблица страниц, ошибка страницы - page fault) и укажите их роль в операционной системе.

17. Проанализируйте механизмы синхронизации (мьютекс, семафор, барьер, spinlock) в параллельном программировании и опишите области их применения.
18. Опишите архитектуру гетерогенных вычислительных систем (включающих CPU, GPU, FPGA, TPU) и их значение для современных систем искусственного интеллекта и Больших данных.
19. Объясните суть модели "3V" (Volume, Velocity, Variety) в концепции Больших данных (Big Data) и ее непосредственную связь с параллельными вычислениями.
20. Какие основные функции и механизмы обмена сообщениями стандарта MPI (Message Passing Interface) в рамках параллельного программирования?
21. Объясните роль и основные возможности технологии OpenMP в организации многопоточного программирования в вычислительных системах с общей памятью.
22. Объясните принцип работы и ключевые преимущества модели SPMD (Single Program, Multiple Data), применяемой в технологии OpenMP.
23. В чем заключается фундаментальная суть технологии CUDA, и каков механизм параллельных вычислений, основанный на взаимодействии между CPU и GPU?
24. Проанализируйте роль метода конвейерной обработки в повышении производительности процессора за счет разделения цикла выполнения команд на этапы.
25. Объясните механизмы суперскалярной архитектуры, предоставляющие возможность параллельно выполнять несколько инструкций в течение одного процессорного такта.
26. Объясните концепцию параллелизма на уровне инструкций (ILP) во взаимосвязи с конвейерными и суперскалярными технологиями обработки.
27. Проанализируйте концепцию параллелизма на уровне данных (DLP) и механизмы ее практического применения в векторных и графических процессорах.
28. Сравните концепции параллельной обработки данных и параллельной обработки задач, указав их фундаментальные различия.
29. Обоснуйте принадлежность процессора Intel x86 к архитектуре CISC и объясните ключевые особенности данной микропроцессорной архитектуры.
30. Объясните понятие многоядерной архитектуры (Multicore Architecture) и ее стратегическое значение в разработке современных вычислительных систем.
31. Какие принципы работы технологии многопоточности (Multithreading), и какова ее роль в повышении вычислительной эффективности процессора?
32. В чем заключается суть технологии Hyper-Threading от компании Intel, и каков механизм ее функционирования на аппаратном уровне?
33. Какой принцип действия технологии Simultaneous Multithreading (SMT) от компании AMD, и в чем ее отличие от иных подходов к реализации многопоточности?
34. Сравните понятия "процесс" (process) и "поток" (thread) в контексте функционирования операционной системы, указав их основные отличия.
35. Объясните сущность, принцип работы и преимущества архитектуры симметричных многопроцессорных систем (SMP).

36. Объясните сущность, механизм управления и принципы распределения задач между процессорами в архитектуре асимметричных многопроцессорных систем (AMP).
37. Объясните принцип функционирования вычислительных систем с общей памятью и их значение для построения многопроцессорных архитектур.
38. Какие основные характеристики архитектуры NUMA (Non-Uniform Memory Access) и каким образом время доступа к памяти зависит от физического расстояния до узла?
39. Опишите гибридную модель параллельного программирования и объясните механизм совместного применения подходов с распределенной и общей памятью в её рамках.
40. Сравните архитектуры CPU и GPU с точки зрения количества вычислительных ядер и их вычислительных возможностей в задачах распараллеливания.
41. Поясните функции и механизмы взаимодействия концептуальных сущностей "Хост" (Host) и "Устройство" (Device) в модели программирования CUDA.
42. Опишите структурный состав потокового мультипроцессора (Streaming Multiprocessor — SM) и его функции в процессе вычислений на GPU.
43. Перечислите основные уровни иерархии памяти GPU и объясните их различия с точки зрения выполняемых функций и скорости доступа к данным.
44. Объясните преимущества использования разделяемой памяти (shared memory) в архитектуре GPU и её влияние на эффективность параллельных вычислений.
45. Объясните проблему чрезмерного переключения контекста, возникающую при инициализации избыточного числа потоков в параллельных программах, и её влияние на общую скорость системы.
46. Объясните суть принципа когерентности кэша (Cache Coherence) и его фундаментальную роль в обеспечении согласованности данных в параллельных системах.
47. Объясните принцип работы механизма когерентности на основе отслеживания (Snooping-based) и процесс мониторинга данных через общую системную шину.
48. Объясните причину возникновения проблемы дисбаланса скоростей между процессором и основной памятью, а также её влияние на общую эффективность системы.
49. Объясните аппаратную роль блока управления памятью (MMU) в процессе прозрачной трансляции виртуальных адресов в физические.
50. Объясните причины возникновения аппаратного прерывания "ошибка страницы" (Page Fault) и алгоритм действий операционной системы по разрешению данной ситуации.
51. Объясните назначение таблицы страниц (Page Table) в системе виртуальной памяти и её роль в отображении адресов и контроле доступа.
52. Объясните основные типы узлов вычислительного кластера (вычислительные узлы и головные узлы/узлы входа) и их соответствующие системные функции.
53. Объясните основную цель стандарта MPI в контексте организации эффективного межпроцессного взаимодействия в распределенных системах.
54. Проанализируйте базовые операции обмена сообщениями (отправка/получение - send/receive) между процессами в технологии распределенных вычислений MPI.

55. Объясните роль концепции согласованности памяти (Memory Consistency) в обеспечении предсказуемого и корректного представления данных параллельным потокам.
56. Объясните основную цель программной синхронизации потоков при многопоточном программировании и ее критическое значение для корректного доступа к общим ресурсам.
57. Объясните алгоритм работы механизма синхронизации Lock/Mutex и его первостепенную роль в обеспечении взаимного исключения (mutual exclusion).
58. Сравните механизмы семафоров и мьютексов, проанализируйте логику их работы и концептуальные отличия в сферах практического применения.
59. Объясните назначение примитива синхронизации «барьер» (barrier) в параллельном программировании и его функцию в координации вычислительных потоков.
60. Опишите конкретные ситуации, в которых использование механизма Spinlock является наиболее эффективным, и объясните его преимущества, связанные с экономией на переключении контекста.
61. Проанализируйте архитектурную суть подхода мелкозернистой синхронизации (Fine-grained synchronization) и его роль в повышении общего уровня распараллеливания алгоритма.
62. Объясните концепцию неблокирующего (Lock-free) программирования и методы реализации в ней безопасной синхронизации исключительно на базе атомарных операций процессора.
63. Объясните механизм функционирования атомарной инструкции CAS (Compare-And-Swap) и ее фундаментальное значение для структур параллельного программирования.
64. Объясните суть так называемой "проблемы АВА", возникающей в Lock-free программировании, и ее негативное воздействие на корректность выполнения алгоритмов CAS.
65. Проанализируйте роль аппаратных атомарных процессорных операций в обеспечении абсолютной целостности данных и предотвращении проблем классической синхронизации.
66. Объясните понятие "гетерогенного параллелизма" и механизм кардинального повышения производительности системы за счет комбинированного распределения задач между устройствами разного типа.
67. Перечислите базовые компоненты, входящие в состав современных гетерогенных вычислительных систем, и объясните выполняемые ими специфические вычислительные роли.
68. Объясните целевое назначение специализированных архитектур (таких как ASIC и TPU) и их неоспоримые преимущества перед универсальными процессорами (CPU) при решении узкоспециализированных задач.
69. Проанализируйте роль центрального процессора в реализации общей логики управления и диспетчеризации задач в рамках гетерогенной связки CPU-GPU.
70. Объясните аппаратные причины, обуславливающие необходимость применения массово-параллельного программирования при обработке графики, и влияние данной парадигмы на общую производительность.

71. Объясните смысловое содержание модели "3V" (Volume, Velocity, Variety) в контексте Больших данных (Big Data) и ее значение для выбора методов параллельной обработки информации.
72. Объясните решающую роль кластерной архитектуры и платформ распределенной обработки (таких как MapReduce и Spark) в обеспечении масштабных параллельных вычислений для систем Big Data.
73. Проанализируйте профильные области применения методов параллельного программирования в цифровой обработке сигналов (DSP) и их влияние на функционирование систем жесткого реального времени.
74. Опишите целевые функции и практическое применение специализированных программных фреймворков (NetworkX, GraphX, CUDA Graphs), реализующих графо-ориентированные параллельные вычисления.
75. Объясните объективные причины критической потребности современных систем искусственного интеллекта и глубокого обучения (Deep Learning) в массивно-параллельных вычислительных архитектурах.
76. Объясните функциональные возможности и архитектурные особенности ведущих программных библиотек (Horovod, TensorFlow, PyTorch), применяемых для эффективного распараллеливания расчетов в Deep Learning.
77. Объясните алгоритмическую роль библиотеки Horovod в процессе синхронизации и распределенного обучения тяжелых нейронных сетей в кластерной GPU-среде.
78. Проанализируйте значимость технологического фреймворка DeepSpeed для эффективной программной оптимизации процессов распределенного обучения и логического вывода (inference) гигантских языковых моделей.
79. Объясните принцип работы подхода параллелизма данных (Data Parallelism) в машинном обучении, детально расписав математический механизм синхронизации градиентов между узлами.
80. Объясните сущность подхода тензорного распараллеливания (Tensor Parallelism), применяемого в критических ситуациях, когда весовые параметры нейронной сети физически не помещаются в локальную видеопамять одного GPU.
81. Укажите и проанализируйте фундаментальные причины необходимости внедрения узкоспециализированных ускорителей (GPU/TPU) для выполнения параллельных расчетов в системах глубокого обучения.
82. Сравните функциональные возможности и стратегии распределенных вычислений высокоуровневых программных библиотек TensorFlow, PyTorch и Horovod при организации кластерного обучения.
83. Объясните фундаментальные архитектурные различия между концепциями параллелизма данных (Data Parallelism) и параллелизма моделей (Model Parallelism), проиллюстрировав их релевантными примерами из области ИИ.
84. Детально проанализируйте алгоритм работы подхода Data Parallelism, охватывающий этапы дробления исходных батчей данных и финальной математической синхронизации расчетных градиентов.
85. Охарактеризуйте роль массивно-параллельных архитектур на базе чипов GPU и TPU в обеспечении и кардинальном повышении скорости обучения сложных архитектур нейронных сетей.

86. Опишите специфику применения технологий параллельных вычислений в сфере потоковой цифровой обработки сигналов (DSP) и их непосредственное влияние на пропускную способность телекоммуникационных систем.
87. Назовите и обоснуйте основные факторы, обуславливающие острую потребность передовых систем искусственного интеллекта в сверхпроизводительных параллельных сопроцессорах.
88. Объясните прикладное значение методов параллельного программирования в таких наукоемких дисциплинах, как биоинформатика, квантовая физика и химия, а также их роль в ускорении выполнения сверхсложных дифференциальных вычислений.
89. Проанализируйте и сопоставьте инструментальные возможности платформ TensorFlow и PyTorch в аспекте организации масштабируемых параллельных вычислений, выделив их сильные стороны.
90. Объясните влияние программных и аппаратных алгоритмов балансировки нагрузки (load balancing) на общую пропускную способность и равномерную утилизацию ядер параллельных систем.
91. Объясните строгие теоретические ограничения, налагаемые законом Амдала (Amdahl's Law) на процесс распараллеливания кода, и их математическое влияние на максимально достижимое ускорение вычислительной системы.
92. Сравните принципы аппаратной работы вычислительных архитектур SIMD и MIMD, подробно объяснив специфику управления потоками данных и приведя примеры типичных сфер применения.
93. Объясните с академической точки зрения принципиальные алгоритмические различия между синхронной и асинхронной моделями выполнения логических задач в рамках параллельного программирования.
94. Проанализируйте логическую суть проблемы расхождения потоков (thread divergence), возникающей внутри варпа при ветвлении кода на GPU, и оцените ее негативное воздействие на утилизацию ядер.
95. Объясните ключевые математические критерии оценки эффективности развертывания параллельных алгоритмов, в частности, формальные показатели ускорения (Speedup) и эффективности (Efficiency).
96. Объясните фундаментальные алгоритмические предпосылки возникновения критического состояния взаимоблокировки (deadlock) в многопоточных приложениях и перечислите превентивные методы ее недопущения.
97. Проанализируйте причины возникновения непредсказуемой проблемы состояния гонки (race condition) при разработке параллельного кода и назовите программные способы ее аппаратной и логической блокировки.
98. Объясните принцип работы архитектурной концепции конвейерного параллелизма (pipeline parallelism) и механизм ее положительного воздействия на повышение общей пропускной способности задачи.
99. Проанализируйте влияние строгой многоуровневой иерархии памяти (memory hierarchy) в современной архитектуре GPU на задержки при выборке данных и общую скорость выполнения параллельных ядер.

100. Опишите структуру иерархии памяти в архитектуре графических процессоров (GPU) и детально проанализируйте зависимость скорости параллельных вычислений от характеристик применяемой памяти.