**Контрольная работа**

Контрольная работа представляет собой опрос. Список вопросов выбирается по последней цифре пароля по 12 вопросов из 120 каждому студенту (например, если у вас пароль заканчивается на цифру 5, то вы выбираете темы под номерами 5, 15, 25, 35, …, 115).

1. Критерии оценки производительности многопроцессорных вычислительных систем.
2. В каких величинах проводится оценки производительности многопроцессорных вычислительных систем.
3. Какие существуют основные тесты производительности компьютеров?
4. В чем заключаются основные способы достижения параллелизма?
5. В чем могут состоять различия параллельных вычислительных сис-тем?
6. Что положено в основу классификации Флинна?
7. В чем особенность кластерных вычислительных систем?
8. Какие преимущества предоставляет пользователям вычислительных систем с суперкомпьютерным уровнем кластерная архитектура .
9. Оценки времени передачи сообщений по Хокни.
10. Закон Амдала и его следствия.
11. В чем отличие кластерных вычислительных систем от "мэйнфреймов"?
12. Какие вычислительные системы относятся к grid системам?
13. Какие принципы технической реализации вычислительных систем (ВС) относятся к базовым?
14. Каким требованиям удовлетворяют вычислительные системы, основанные на принципах модульности и близкодействия?
15. Какие архитектурные свойства присущи вычислительным системам?
16. Назовите основополагающие принципы в основе параллельных вычислительных систем.
17. Назовите важнейшие свойства архитектуры ВС.
18. Чем достигается децентрализованность управления ВС?
19. Что понимается под масштабируемостью (Scalability) вычислительных систем?
20. Назовите основные виды реконфигурируемость (Programmability) вычислительных систем и их свойства.
21. Этапы разработки параллельных алгоритмов
22. В чем состоит принцип разделения многопроцессорных систем на мультипроцессоры и мультикомпьютеры?
23. Какие классы систем известны для мультипроцессоров?
24. Какое ускорение вычислений может достигаться для процессоров с поддержкой одновременной многопотоковости?
25. Как определяется расписание для распределения вычислений между процессорами?
26. Как определяется время выполнения параллельного алгоритма?
27. Какое расписание является оптимальным?
28. Как определить минимально возможное время решения задачи?
29. Что понимается под паракомпьютером и для чего может оказаться полезным данное понятие?
30. Какие оценки следует использовать в качестве характеристики времени последовательного решения задачи?
31. Как определить минимально возможное время параллельного решения задачи по графу «операнды–операции»?
32. Какие зависимости могут быть получены для времени параллельного решения задачи при увеличении или уменьшения числа используемых процессоров?
33. При каком числе процессоров могут быть получены времена вы-полнения параллельного алгоритма, сопоставимые по порядку с оценками минимально возможного времени решения задачи?
34. Как определяются понятия ускорения и эффективности?
35. Возможно ли достижений сверхлинейного ускорения?
36. В чем состоит противоречивость показателей ускорения и эффек-тивности?
37. Как определяется понятие стоимости вычислений?
38. В чем состоит понятие стоимостно-оптимального алгоритма?
39. В чем заключается проблема распараллеливания последователь-ного алгоритма суммирования числовых значений?
40. Какие основные характеристики используются для оценки топологии сети передачи данных? Приведите значения характеристик для конкретных типов коммуникационных структур (полный граф, линейка, решетка и др.).
41. Какие основные методы применяются при маршрутизации передаваемых данных по сети?
42. В чем состоят основные методы передачи данных? Приведите для этих методов аналитические оценки времени выполнения.
43. Какие операции передачи данных могут быть выделены в качестве основных?
44. В чем состоят алгоритмы выполнения передачи данных от одного процессора всем процессорам сети для топологий кольца, решетки и гиперкуба? Приведите оценки временной трудоемкости для этих алгоритмов.
45. В чем состоят алгоритмы выполнения передачи данных от всех процессоров всем процессорам сети для топологий кольца, решетки и гиперкуба? Приведите оценки временной трудоемкости для этих алгоритмов.
46. Имеют ли подзадачи после масштабирования одинаковую вычислительную и коммуникационную сложность?
47. Зависят ли параметрически правила масштабирования от количества процессоров?
48. Что означает функциональный параллелизм?
49. Какие регулярные структуры базовых подзадач получаются после декомпозиции данных?
50. Какие компьютерные платформы относятся к числу вычислительных систем с общей памятью?
51. Какие подходы используются для разработки параллельных программ?
52. В чем состоят основы технологии OpenMP?
53. В чем состоит важность стандартизации средств разработки параллельных программ?
54. В чем состоят основные преимущества технологии OpenMP?
55. Что понимается под параллельной программой в рамках технологии OpenMP?
56. Что понимается под понятием потока (thread)?
57. Какие проблемы возникают при использовании общих данных в  
    параллельно выполняемых потоках?
58. Какой формат записи директив OpenMP?
59. В чем состоит назначение директивы **parallel?**
60. В чем состоят понятия фрагмента, области и секции параллельной программы?
61. Какой минимальный набор директив OpenMP позволяет начать разработку параллельных программ?
62. Как определить время выполнения OpenMP программы?
63. Как осуществляется распараллеливание циклов в OpenMP? Какие условия должны выполняться, чтобы циклы могли быть распараллелены?
64. Какие возможности имеются в OpenMP для управления распреде­лением итераций циклов между потоками?
65. Как определяется порядок выполнения итераций в распараллеливаемых циклах в OpenMP?
66. Какие правила синхронизации вычислений в распараллеливаемых циклах в OpenMP?
67. Как определяются общие и локальные переменные потоков?
68. Что понимается под операцией редукции?
69. Какие способы организации взаимоисключения могут быть ис­пользованы в OpenMP?
70. Что понимается под атомарной (неделимой) операцией?
71. Как определяется критическая секция?
72. Какие операции имеются в OpenMP для переменных семафорного типа (замков)?
73. В каких ситуациях следует применять барьерную синхрониза­цию?
74. Как осуществляется в OpenMP распараллеливание по задачам (директива **sections)?**
75. Как определяются однопотоковые участки параллельных фраг­ментов (директивы **single** и **master)?**
76. Как осуществляется синхронизация состояния памяти (директива

**flush)?**

1. Как используются постоянные локальные переменные потоков (директивы **threadprivate** и **copyin)?**
2. Какой минимальный набор функций MPI позволяет начать разработку параллельных программ?
3. Как определить время выполнения MPI программы?
4. В чем различие парных и коллективных операций передачи данных?
5. Какая функция MPI обеспечивает передачу данных от одного процесса всем процессам?
6. Какие режимы передачи данных поддерживаются в MPI?
7. Как организуется неблокирующий обмен данными в MPI?
8. В чем состоит понятие тупика? В каких ситуациях функция одновременного выполнения передачи и приема гарантирует отсутствие тупиковых ситуаций?
9. Какие коллективные операции передачи данных предусмотрены в MPI?
10. Что понимается под производным типом данных в MPI?
11. Какие способы конструирования типов имеются в MPI?
12. Что понимается в MPI под коммуникатором?
13. Что понимается в MPI под виртуальной топологией?
14. Какие виды топологий предусмотрены в MPI?
15. Напишите программу вывода натурального числа в параллельной области всеми нитями; главной нитью; любой нитью.
16. Напишите программу суммирования ряда из n натуральных чисел. Если n<500 программа должна выполняться последовательно, если n=>500 программа должна выполняться параллельно (применение опции if директивы parallel).
17. Напишите программу скалярного произведения двух векторов. При инициализации параллельной области явно укажите количество используемых нитей 4.
18. Напишите программу скалярного произведения двух векторов. При инициализации параллельной области явно укажите количество используемых нитей 4. Вывод результата скалярного произведения произвела главная нить; первая, завершившая вычисления нить.
19. Дан одномерный массив A из n элементов, расположенный в общей памяти. Написать программу подсчет суммы элементов массива A. (Примечание: использование опции reduction директивы parallel).
20. Дан одномерный массив A из n элементов, расположенный в общей памяти. Написать программу определения маскимального и минимального элементов массива A. (Примечание: использование опции reduction директивы parallel).
21. Дан одномерный массив A из n элементов, расположенный в общей памяти. Написать программу, в которой нити с четными номерами выводят на экран четные элементы массива A, а нити с нечетными номерами выводят на экран нечетные элементы массива A. (Примечание: использование опции if директивы parallel).
22. Дан одномерный массив A из n элементов, расположенный в общей памяти. Написать программу подсчет суммы элементов массива A. Если n меньше 100, то программа выполняется последовательно только главной нитью. Если n больше 100, то подсчет суммы выполняется всеми нитями. Если n равно 100, то программа аварийно завершается. (Примечание: использование опций if, reduction директивы parallel).
23. Дан одномерный массив A из n элементов, расположенный в общей памяти. Написать программу подсчет суммы элементов массива A, количество нитей задается опцией num\_threads. (Примечание: использование опций num\_threads, reduction директивы parallel).
24. Дан одномерный массив A из n элементов, расположенный в общей памяти. Написать программу подсчета количества нулевых элементов массива A. (Примечание: использование опции reduction директив parallel, for (do)).

101–104. Написать и реализовать параллельную программу на 4-х ядерном процессоре умножения квадратной матрицы на вектор при разделении данных по строкам. Размерность массивов N = № пароля \*100. Использовать OpenMP.

105–107. Написать и реализовать параллельную программу перемножения квадратных матриц на 4-х ядерном процессоре. Размерность массивов N = № пароля \*100. Использовать OpenMP.

108–110. Написать и реализовать программу параллельного решения систем линейных уравнений методом Гаусса на 4-х ядерном процессоре. Размерность массивов N = № пароля \*100.Использовать OpenMP.

111–115. Написать программу параллельного решения СЛАУ методом Гаусса. Распределение данных - горизонтальными полосами. Запуск задачи на 8-ми компьютерах. Размерность массивов N = № пароля \*100. Использовать MPI.

116–120. Написать программу параллельного решения задачи Дирихле методом Гаусса-Зейделя для 4-х процесcорной ВС. Размерность массивов N = № пароля \*100. Использовать MPI.