

Паспорт экзамена

по дисциплине «Программное обеспечение высокопроизводительных вычислений», 2
семестр

1. Методика оценки

Экзамен проводится в устной форме, по билетам. Билет формируется по следующему правилу: первый вопрос выбирается из диапазона вопросов 1-28, второй вопрос из диапазона вопросов 29-57 (список вопросов приведен ниже). В ходе экзамена преподаватель вправе задавать студенту дополнительные вопросы из общего перечня (п. 4).

Форма экзаменационного билета

НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
Факультет АВТФ

Билет № _____

к экзамену по дисциплине «Программное обеспечение высокопроизводительных
вычислений»

1. Вопрос 1. Модели многопроцессорных систем с общей и распределенной памятью. Модель конвейерной системы.
2. Вопрос 2. Распараллеливающие компиляторы, проблема выделения потенциального параллелизма последовательных программ. Специальные комментарии и директивы компилятору.

Утверждаю: зав. кафедрой _____ должность, ФИО
(подпись) (дата)

2. Критерии оценки

- Ответ на экзаменационный билет считается **неудовлетворительным**, если студент при ответе на вопросы не дает определений основных понятий, не способен показать причинно-следственные связи явлений, оценка составляет 0-49 *баллов*.
- Ответ на экзаменационный билет засчитывается на **пороговом** уровне, если студент при ответе на вопросы дает определение основных понятий, может показать причинно-следственные связи явлений, оценка составляет 50-72 *баллов*.
- Ответ на экзаменационный билет засчитывается на **базовом** уровне, если студент при ответе на вопросы формулирует основные понятия, законы, дает характеристику процессов, явлений, проводит анализ причин, условий, может представить

качественные характеристики процессов, оценка составляет 73-86 *баллов*.

- Ответ на экзаменационный билет засчитывается на **продвинутом** уровне, если студент при ответе на вопросы приводит сравнительный анализ подходов, проводит комплексный анализ, выявляет проблемы, предлагает механизмы решения, способен представить количественные характеристики определенных процессов, приводит конкретные примеры из практики, оценка составляет 87-100 *баллов*.

3. Шкала оценки

В общей оценке по дисциплине экзаменационные баллы учитываются в соответствии с правилами балльно-рейтинговой системы, приведенными в рабочей программе дисциплины:

Балл	Оценка ECTS	Оценка
98 – 100	A+	5
93 – 97	A	
90 – 92	A–	
87 – 89	B+	
83 – 86	B	4
80 – 82	B–	
77 – 79	C+	
73 – 76	C	
70 – 72	C–	3
67 – 69	D+	
63 – 66	D	
60 – 62	D–	
50 – 59	E	
Меньше 50	F	2

4. Вопросы к экзамену по дисциплине «Программное обеспечение высокопроизводительных вычислений»

1. Динамические свойства параллельных программ.
2. Постановка задачи сбалансированного распределения вычислительной
3. загрузки на мультикомпьютерах.
4. Обзор методов и программных средств разбиения графов.
5. Обзор системы NumGRID
6. Классификация распределенных вычислительных систем.
7. Требования к организации грид-систем.
8. Каковы ограничения максимальной производительности однопроцессорных ЭВМ.
9. Параллельные и распределенные вычисления и их техническая основа - вычислительные кластеры, ГРИД-системы и суперкомпьютеры.
10. Виды параллелизма.
11. Основные проблемы использования параллельной обработки данных.
12. Закон Амдаля о существовании последовательных алгоритмов.
13. Закон Мура о росте производительности последовательных компьютеров.
14. Закон Гроша о стоимости параллельных систем.

15. Гипотеза Минского о влиянии потерь на взаимодействие на степень ускорения параллельных вычислений по сравнению с последовательными.
16. Конвейерные и векторные вычисления. Процессорные матрицы. Многопроцессорные вычислительные системы с общей и распределенной памятью.
17. Схемы коммутации.
18. Схемы взаимодействия ветвей параллельных алгоритмов
19. Типовые топологии схем коммутации.
20. Аппаратная реализация и программная эмуляция топологий.
21. Классификация многопроцессорных вычислительных систем.
22. Систематика Флинна. Потоки данных (команд).
23. Виды многопроцессорных систем и кластеров
24. Модели параллельных вычислительных процессов.
25. Концепция неограниченного параллелизма. Компьютер с неограниченным параллелизмом (паракомпьютер).
26. Модели многопроцессорных систем с общей и распределенной памятью. Модель конвейерной системы.
27. Модель алгоритма в виде графа "операнд - операции". Представление алгоритма в виде графа потока данных.
28. Расписание параллельных вычислений. Показатель временной сложности алгоритма.
29. Оценка времени выполнения алгоритма для паракомпьютера (предельное распараллеливание) и для систем с конечным количеством процессоров.
30. Способы получения оптимального расписания вычислений.
31. Модель параллельных вычислений в виде сети Петри.
32. Основные проблемы параллельных вычислений: синхронизация, взаимное исключение, блокировка (тупики).
33. Поточковая модель параллельных вычислений.
34. Проблемы взаимодействия процессов. Синхронизация параллельных процессов.
35. Механизмы взаимного исключения: алгоритм Деккера, семафоры (Дейкстра), мониторы (Вирт).
36. Взаимодействие параллельных процессов посредством механизма передачи сообщений. Механизмы передачи.
37. Понятие тупика и условия его возникновения. Предотвращение тупиков. Обнаружение тупиков и восстановление состояния процессов.
38. Параллелизм данных и параллелизм задач. Показатель эффективности распараллеливания (ускорение).
39. Эффективность использования вычислительной системы. Способы оценки показателей.
40. Оценка коммуникационной трудоемкости параллельных алгоритмов. Характеристики топологий сети передачи данных.
41. Алгоритмы маршрутизации. Методы передачи данных. Анализ трудоемкости основных операций передачи данных.
42. Передача данных между двумя процессорами сети.

43. Одиночная и множественная рассылка сообщений. Операция циклического сдвига.
44. Уровни распараллеливания вычислений: команд, выражений, программных модулей, отдельно выполняемых заданий.
45. Этапы построения параллельных алгоритмов и программ.
46. Реализация алгоритма в виде параллельной программы. Построение исполняемой программы для параллельной вычислительной системы.
47. Использование распространенных языков программирования и коммуникационных библиотек и интерфейсов.
48. Распараллеливающие компиляторы, проблема выделения потенциального параллелизма последовательных программ. Специальные комментарии и директивы компилятору.
49. Параллельные языки программирования и расширения стандартных языков.
50. Средства автоматического распараллеливания, параллельные компиляторы.
51. Параллельные предметные библиотеки. Инструментальные системы для проектирования параллельных программ.
52. Предотвращение тупиков. Коллективные функции обмена данных: широковещательная рассылка, функции сбора и рассыпания данных.
53. Функции редукции данных. Создание групп процессов, области связи, коммутаторы. Обмен данными внутри группы, межгрупповой обмен.
54. Топология обменов. Декартовы топологии. Топологии произвольного графа.
55. Последовательные и параллельные нити программы. Организация параллельных секций.
56. Параллельные циклы. Директивы синхронизации. Классы переменных.
57. Перспективы развития МВС и параллельного программирования.