

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
«НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Факультет прикладной математики и информатики

“УТВЕРЖДАЮ”

Декан ФПМИ

профессор, д.т.н. Лемешко
Борис Юрьевич

“ ____ ” _____ г.

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ

Основы параллельного программирования

ООП: направление 010500.62 Прикладная математика и информатика

Шифр по учебному плану: СД(М).Ф.6

Факультет: прикладной математики и информатики очная форма обучения

Курс: 4, семестр: 8

Лекции: 36

Практические работы: - Лабораторные работы: 48

Курсовой проект: - Курсовая работа: - РГЗ: -

Самостоятельная работа: 41

Экзамен: 8 Зачет: -

Всего: 125

Новосибирск

2011

Рабочая программа составлена на основании Государственного образовательного стандарта высшего профессионального образования по направлению (специальности): 510200 Прикладная математика и информатика.(№ 200 ен/бак от 23.03.2000)

СД(М).Ф.6, дисциплины федерального компонента

Рабочая программа обсуждена на заседании кафедры Параллельные вычислительные технологии протокол № 79 от 24.06.2011

Программу разработал

доцент, к.т.н.

Корнеев Владимир Дмитриевич

Заведующий кафедрой

профессор, д.т.н.

Малышкин Виктор Эммануилович

Ответственный за основную образовательную программу

профессор, д.т.н.

Соловейчик Юрий Григорьевич

1. Внешние требования

Таблица 1.1

Шифр дисциплины	Содержание учебной дисциплины	Часы
СД(М).Ф.6	Методы и средства параллельной обработки информации: параллельные вычислительные методы, параллельные вычислительные системы, параллельное программирование; векторизация последовательных выражений алгоритмов, методология канонического отображения алгоритма в графы зависимостей и потока сигналов, в матричный процессор; средства спецификации параллельных процессов; механизмы взаимодействия асинхронных параллельных процессов; синхронизирующие примитивы; методы и языки параллельного программирования: язык MPI, язык OpenMP, язык потоков данных; основные конструкции и приемы программирования; применения языков для решения практических задач; сравнение языков; эффективность применения; мультипрограммные системы; параллельная обработка информации в мультисистемах. 1.3. Квалификационные требования Бакалавр прикладной математики и информатики подготовлен преимущественно к выполнению исследовательской деятельности в областях, использующих методы прикладной математики и компьютерные технологии; к разработке и применению современных математических методов и программного обеспечения для решения задач науки, техники, экономики и управления; к использованию информационных технологий в проектно-конструкторской, управленческой и финансовой деятельности. 7.1. Требования к профессиональной подготовленности специалиста Бакалавр прикладной математики и информатики: должен обладать: теоретическими знаниями и практическими навыками, соответствующими основной образовательной программе подготовки настоящего государственного образовательного стандарта.	125

	должен знать и уметь использовать: архитектурные особенности современных ЭВМ; конструкции распределенного и параллельного программирования; иметь опыт: работы на различных типах ЭВМ, применения стандартных алгоритмических языков, использования приближенных методов и стандартного программного обеспечения для решения прикладных задач. должен: обладать знаниями и умениями, позволяющими применять современные математические методы и программное обеспечение для решения задач науки, техники, экономики и управления и использования информационных технологий в проектно-конструкторской, управленческой и финансовой деятельности. быть способен к совершенствованию своей профессиональной деятельности в области прикладной математики и информатики.	
--	---	--

2. Особенности (принципы) построения дисциплины

Таблица 2.1

Особенности (принципы) построения дисциплины

Особенность (принцип)	Содержание
Основания для введения дисциплины в учебный план по направлению или специальности	Стандарт направления 510200, решение ученого Совета ФПМИ от 24.06.2003, протокол № 6.
Адресат курса	Студенты по направлению 010500 - "Прикладная математика и информатика".
Основная цель (цели) дисциплины	Обеспечение базы теоретической и практической подготовки в области параллельного программирования, развитие мышления связанного с параллельными вычислениями, приобретение знаний, необходимых для изучения последующих дисциплин
Ядро дисциплины	Модели и методы параллельного программирования, типовые параллельные программы
Связи с другими учебными дисциплинами основной образовательной программы	Практикум на ЭВМ, архитектура ЭВМиВС, выпускная квалификационная работа.

Требования к первоначальному уровню подготовки обучающихся	Необходимы знания по вычислительной математике. Необходимы знания языка программирования C. Опыт работы на персональном компьютере, знание некоторых прикладных программ.
Особенности организации учебного процесса по дисциплине	Практическая часть дисциплины содержит лабораторные работы. Студенты осваивают средства и методы параллельного программирования на вычислительных кластерах. Часть заданий меняется. Для проведения лабораторных работ используются методические указания, как в печатной, так и в электронной форме по параллельному программированию в MPI и OpenMP. Современные информационные технологии MPI и OpenMP являются предметом исследования и освоения, а так же для чтения лекций и проведения практических занятий используется PowerPoint

3. Цели учебной дисциплины

Таблица 3.1

После изучения дисциплины студент будет

иметь представление	
1	О параллельных программах и алгоритмах
5	О средствах параллельного программирования на системах с распределенной и общей памятью
6	О методах параллельного программирования
7	О моделях параллельно-последовательного программирования
8	О моделях асинхронных вычислений
9	О моделях синхронных вычислений
17	О проблемах и направлениях развития технологии программирования
знать	
2	Средства параллельного программирования MPI
10	Средства параллельного программирования OpenMP
11	Методы распараллеливания задач линейной алгебры
12	Методы распараллеливания задач решаемых сеточными методами
18	Модели параллельных алгоритмов и методы параллельного программирования
19	Механизмы и средства параллельного программирования
24	Основы теории алгоритмов.
уметь	
3	Создавать параллельные программы для алгоритмов матричных задач
13	Создавать параллельные программы для задач решения систем линейных алгебраических уравнений как прямыми, так и итерационными методами
14	Создавать параллельные программы для алгоритмов матричных задач на вычислительных системах с топологией двух и трех мерных решеток
15	Создавать параллельные программы для задач, решаемых сеточными методами
20	Создавать математические и алгоритмические модели
21	Применять языки программирования для решения практических задач
иметь опыт (владеть)	
4	Практического распараллеливания алгоритмов реальных задач
16	Практической работы на современных вычислительных кластерах
22	Построения основных моделей информационных технологий и способов их применения для решения задач в предметных областях
23	Выбора архитектуры современных компьютеров и систем
25	Применения стандартных алгоритмических языков, использования стандартного программного обеспечения, пакетов прикладных программ для решения прикладных задач.

4. Содержание и структура учебной дисциплины

Лекционные занятия

Таблица 4.1

(Модуль), дидактическая единица, тема	Часы	Ссылки на цели
Семестр: 8		
Модуль: Общее представление алгоритма, параллельного алгоритма		
Дидактическая единица: Мультипрограммные системы		
Представление алгоритма. Определение алгоритма (по ак. Мальцеву А.Н.). Требования к представлению параллельного алгоритма. Машина Тьюринга.	2	1, 24
Модуль: Общая модель алгоритма: Сети Петри		
Дидактическая единица: Методология канонического отображения алгоритма в графы зависимостей и потока сигналов		
Определение сети Петри. Разметка сети. Срабатывание переходов (правила). Примеры сетей Петри. Неограниченная сеть. Ограниченная сеть, но не безопасная (тупиковая). Моделирование ограниченного буфера. Отображение алгоритма в графы зависимостей.	4	1, 20, 24, 8
Модуль: Взаимодействующие процессы		
Дидактическая единица: Параллельные вычислительные методы; параллельные вычислительные системы; параллельное программирование.		
Синхронизация процессов через доступ к общим ресурсам. Критические секции. Семафоры. Мониторы.	10	18, 5, 6, 7, 8, 9
Синхронизация процессов через обмен данными. Асинхронные взаимодействия. Синхронные взаимодействия.	4	1, 18, 6, 8, 9
Модуль: Параллельно-последовательные модели программирования		
Дидактическая единица: Механизмы взаимодействия в параллельно-последовательных моделях программирования		
Ярусно-параллельные формы	1	18, 7
MPMD - модель	1	18, 7
SPMD - модель	2	18, 7
Модуль: Модели асинхронных вычислений		
Дидактическая единица: Методы параллельных асинхронных вычислений; механизмы взаимодействия асинхронных параллельных процессов.		
Событийное управление	2	1, 18, 8
Потоковое управление	2	1, 18, 8
Динамическое управление	2	1, 18, 8
Модуль: Синхронные вычисления		
Дидактическая единица: Методы параллельных		

синхронных вычислений; синхронизирующие примитивы; отображение алгоритма в матричный процессор; язык потоков данных; язык Оккам.		
Примеры синхронных вычислений, матричные вычисления, вычисления на языке Оккам	2	1, 18, 9
Модуль: Языки параллельного программирования		
Дидактическая единица: Средства спецификации параллельных процессов; основные конструкции и приемы параллельного программирования; применения языков для решения практических задач; сравнения языков; эффективность применения.		
Язык MPI	2	1, 11, 12, 13, 14, 15, 16, 2, 3, 4, 5
Язык OpenMP	1	1, 10, 11, 12, 13, 14, 15, 3, 5
язык ОККАМ, язык HPF.	1	1, 11, 13, 14, 15, 16, 17, 22, 23, 5

Лабораторная работа

Таблица 4.2

(Модуль), дидактическая единица, тема	Учебная деятельность	Часы	Ссылки на цели
Семестр: 8			
Модуль: Общее представление алгоритма, параллельного алгоритма			
Дидактическая единица: Средства спецификации параллельных процессов в MPI.			
Создание простейших параллельных программ, компиляция и запуск параллельных программ	Получает представление о параллельных программах для систем с распределенной памятью; Обучается компилировать и запускать параллельные программы на системах с распределенной памятью.	2	2
Модуль: Общая модель алгоритма: Сети Петри			
Дидактическая единица: Базовые модели распределенных			

вычислений. Механизмы взаимодействия параллельных процессов в MPI.			
Создание параллельных программ: обмен данными между двумя параллельными ветвями, создание виртуальных топологий	Получает первые навыки написания параллельных программ в системе MPI; Изучает функции обмена данными в системе MPI.	2	2
Модуль: Взаимодействующие процессы			
Дидактическая единица: Методы программирования на системах с распределенной памятью матричных умножений; приемы программирования в MPI матричных умножений.			
Создание параллельных программ алгоритма умножения матрицы на вектор в двух вариантах: в топологии "кольцо" и "полный граф"	Обучается создавать виртуальные топологии в системе MPI; Изучает обменные функции в системе MPI.	4	11, 2, 3, 5
Дидактическая единица: Методы распараллеливания на системах с распределенной памятью задач линейной алгебры.			
Создание параллельных программ алгоритма умножения матрицы на матрицу в двух вариантах: в топологии "кольцо" и "полный граф"	Углубляет знания по созданию виртуальных топологий в MPI; Обучается создавать простые параллельные программы для матричных задач в MPI.	4	11, 2, 3, 4, 5, 6
Дидактическая единица: Модель вычислений над общим полем памяти. Механизмы взаимодействия, синхронизации параллельных процессов в OpenMP.			
Создание параллельных программ алгоритма умножения матрицы на вектор и матрицы на матрицу	Углубляет знания по созданию параллельных программ для матричных задач в MPI.	4	10, 11, 4
Модуль: Параллельно-последовательные модели			

программирования			
Дидактическая единица: Методы и приемы программирования в MPI параллельных алгоритмов решения систем линейных алгебраических уравнений итерационными методами.			
Создание параллельных программ для решения систем линейных алгебраических уравнений методом простой итерации в двух вариантах	Получает представление о параллельных программах для систем с общей памятью; Обучается компилировать и запускать параллельные программы на системах с общей памятью; Получает первые навыки написания параллельных программ в системе OpenMP; Изучает основные директивы в системе OpenMP.	2	11, 13, 2, 4
Дидактическая единица: Приемы и методы программирования в MPI параллельных алгоритмов решения СЛАУ прямыми методами.			
Создание параллельных программ для решения систем линейных алгебраических уравнений методом Гаусса в двух вариантах: с горизонтальным распределением полос матриц коэффициентов и горизонтально-циклическим	Обучается писать программы для решения СЛАУ итерационными методами в системах с общей памятью.	6	11, 13, 16, 2, 4, 5, 6
Дидактическая единица: Приемы и методы программирования на системах с распределенной памятью не вычислительных задач.			
Создание параллельных программ сортировки множеств	Обучается писать программы для решения СЛАУ методом Гаусса в MPI; Обучается понимать разницу между параллельными алгоритмами в зависимости от способа распределения данных	2	16, 2, 4

	по компьютерам в MPI.		
Модуль: Модели асинхронных вычислений			
Дидактическая единица: Методы и приемы программирования над общим полем памяти параллельных алгоритмов решения СЛАУ итерационными методами.			
Создание параллельных программ в OpenMP для решения систем линейных алгебраических уравнений методом простой итерации	Обучается писать программы для решения СЛАУ итерационными методами в распределенных системах.	2	10, 11, 13, 4
Модуль: Синхронные вычисления			
Дидактическая единица: Методы программирования в MPI задач линейной алгебры на Декартовых структурах мультимикомпьютеров.			
Создание параллельных программ алгоритма умножения матрицы на матрицу: в топологии "двумерная решетка"	Обучается писать параллельные программы для сортировок множеств в системе MPI.	6	11, 14, 16, 2, 3, 4
Дидактическая единица: Методы параллельного программирования алгоритмов задач, решаемых сеточными методами.			
Создание параллельных программ для решения уравнений Пуассона в двух и трехмерных областях	Углубляет знания по созданию виртуальных топологий в системе MPI; Обучается создавать MPI-типы данных, обмен MPI-типами.	6	12, 15, 16, 18, 2, 22, 4, 5, 6
Модуль: Языки параллельного программирования			
Дидактическая единица: Основные конструкции и методы программирования алгоритмов решения практических задач математической физики на мультимикомпьютерах и многоядерных вычислительных системах.			
Распараллеливание алгоритмов решения задач математической физики в системе MPI + OpenMP. Вычисление обыкновенных интегралов, решение задачи	Обучается создавать программы для задач, решаемых сеточными методами.	8	10, 15, 16, 2, 20, 21, 22, 25, 4, 5, 6

Дирихле в 2D и 3D.			
--------------------	--	--	--

5. Самостоятельная работа студентов

Семестр- 8, Подготовка к занятиям

Перед каждой лабораторной работой студенты должны внимательно прочитать и изучить соответствующую тему по учебному пособию:

"Параллельное программирование кластеров" / В. Д. Корнеев ; Новосиб. гос. техн. ун-т. - Новосибирск, 2008. - 310, [1] с. : ил. смотри в п. 8.1

и по методическому пособию смотри в п. 8.1: методические указания по лабораторным по курсу "Основы параллельного программирования".

Подготовка к занятиям в течение семестра состоит в подготовке к лабораторным работам:

- 1) Изучение литературы по теме лабораторных работ -- 10 часов.
- 2) Анализ и проработка указанных преподавателем примеров
в учебном пособии по теме лабораторных работ. -- 6 часов.
- 3) Завершение лабораторных работ, начатых на аудиторном занятии -- 18 часов.
- 4) Оформление пояснительной записки к проделанной лабораторной работе -- 7 часов.
 - а) Описание решаемой задачи.
 - б) Графики сравнительных характеристик параллельного и последовательного алгоритмов.
 - в) Текст отлаженной параллельной программы.

Итого: 41 час.

Программы параллельных алгоритмов решаемых задач реализуются на параллельной вычислительной системе НГТУ и ИВМиМГ СО РАН.

6. Правила аттестации студентов по учебной дисциплине

Для аттестации студентов по дисциплине используется балльно-рейтинговая система. Рейтинг студента по дисциплине определяется как сумма баллов за работу в семестре (текущий рейтинг) и баллов, полученных в результате итоговой аттестации (экзамен).

В таблице 6.1 приведено максимальное количество баллов, которое может набрать студент по видам учебной деятельности в течение семестра и диапазоны баллов, соответствующие минимальному и максимальному количеству баллов. Максимальная сумма баллов за семестр составляет 100 баллов (текущий рейтинг - 60 баллов, итоговая аттестация - 40 баллов).

Правила текущей аттестации:

1. В течение восьмого семестра необходимо представить и защитить 12 лабораторных работ в сроки, установленные учебным графиком (см. таблицу 6.1).
2. На защите лабораторной работы предлагается 4-6 вопросов (по ходу выполнения работы). Вопросы смотри в пункте 8.1.
3. Максимальное количество баллов (4 или 6, 8 в зависимости от вида работы) выставляется, если студент полностью ответил на все вопросы, без серьезных замечаний и недочетов.
4. Количество баллов 3, 5 или 7 (в зависимости от вида работы) выставляется, если студент полностью ответил на пять вопросов из шести.
5. Минимальное количество баллов 2 или 4, 6 (в зависимости от вида работы) выставляется, если студент ответил на четыре вопроса из шести частично, с серьезными замечаниями, недочетами.
6. Пересдача лабораторной работы назначается, если студент не ориентируется в учебном материале, не может объяснить ход и результаты выполнения работы. В случае пересдачи работы происходит потеря баллов (максимальное количество баллов составляет 3 или 6 в зависимости от вида работы).
7. В случае представления и защиты работ с опозданием от учебного графика происходит потеря баллов (опоздание на 1 неделю - потеря 1 или 2 балла в зависимости от вида работы, опоздание на 2 недели - потеря 2 или 3 балла, 3 недели и более - потеря 50% баллов от максимально возможного).

Правила итоговой аттестации:

1. К экзамену допускаются студенты, сдавшие лабораторные работы и набравшие не менее 50% (30 баллов) по результатам текущего рейтинга.
2. Экзамен проводится в письменном виде, предлагается билет из двух теоретических вопросов и одной задачи (образец экзаменационного билета приведен в приложении). Теоретические вопросы смотри в пункте 9 рабочей программы.
3. Максимальное количество 40 баллов выставляется, если все задания выполнены полностью, без серьезных замечаний.
4. Количество баллов 30-35 выставляется, если успешно выполнены два задания из трех.
5. Минимальное количество баллов 20-29 выставляется, если выполнены два задания из трех, но с серьезными ошибками, замечаниями, недочетами.
6. Возможно получить «автомат» (отлично) по дисциплине без сдачи экзамена, если студент в течение семестра выполняет дополнительные задания повышенной сложности и набирает свыше 90 баллов по текущему рейтингу.

К дополнительным заданиям повышенной сложности относится самостоятельное углубленное освоение и выполнение более сложных лабораторных работ.

За дополнительные задания максимально выставляется 40 баллов.

Таблица 6.1

№п/п	Вид учебной работы	Макс. колич. баллов	Диап. баллов	Сроки защиты
Восьмой семестр:				
1.	Лабораторная работа №1	4	2-4	5
2.	Лабораторная работа №2	4	2-4	6
3.	Лабораторная работа №3	4	2-4	7
4.	Лабораторная работа №4	4	2-4	8
5.	Лабораторная работа №5	4	2-4	9
6.	Лабораторная работа №6	4	2-4	10
7.	Лабораторная работа №7	4	2-4	11
8.	Лабораторная работа №8	4	2-4	12
9	Лабораторная работа №9	6	3-6	13
10	Лабораторная работа №10	6	3-6	14
11	Лабораторная работа №11	8	4-8	15
12	Лабораторная работа №12	8	4-8	17
Итого по текущему рейтингу:		60	30-60	
9.	Экзамен	40	20-40	
Итого за восьмой семестр:		100		

97-100 > A+	70-73 => C-	87-100 (отл.)
94-96 => A	74-76 => C	74-86 (хор.)
90-93 => A-	67-69 => D+	50-73 (уд.)
87-89 => B+	64-66 => D	30-50 (не уд.)
84-86 => B	60-63 => D-	
80-83 => B-	50-59 => E	
77-79 => C+		

7. Список литературы

7.1 Основная литература

В печатном виде

1. Корнеев В. Д. Параллельное программирование кластеров : учебное пособие / В. Д. Корнеев ; Новосиб. гос. техн. ун-т. - Новосибирск, 2008. - 310, [1] с. : ил.

7.2 Дополнительная литература

В печатном виде

1. Малышкин В. Э. Параллельное программирование мультимикомпьютеров : [учебник] / В. Э. Малышкин, В. Д. Корнеев. - Новосибирск, 2006. - 295 с. : ил.
2. Эндрюс Г. Д. Основы многопоточного, параллельного и распределенного программирования : [пер. с англ.] / Грегори Р. Эндрюс. - М., 2003. - 505 с. : ил.
3. Воеводин В. В. Параллельные вычисления : [учебник для вузов по направлению 510200 "Прикладная математика и информатика"] / В. В. Воеводин, Вл. В. Воеводин. - СПб., 2004. - 599 с. : ил. - Рекомендовано МО.
4. Немнюгин С. А. Параллельное программирование для многопроцессорных вычислительных систем / Сергей Немнюгин, Ольга Стесик. - СПб., 2002. - 396 с. : ил.
5. Богачев К. Ю. Основы параллельного программирования / К. Ю. Богачёв. - М., 2003. - 342 с.
6. Корнеев В. Д. Параллельное программирование в MPI / В. Д. Корнеев. - М., 2003. - 303 с. : ил.
7. Корнеев В. Д. Параллельное программирование в MPI / Отв. ред. В. Э. Малышкин. - Новосибирск, 2002. - 214 с. : ил.
8. Кнут Д. Э. Искусство программирования для ЭВМ. Т. 3. Сортировка и поиск / Д. Кнут ; под ред.: Ю. М. Баяковского, В. С. Штаркмана. - М., 1978. - 844 с. : ил.
9. Малышкин В. Э. Основы параллельных вычислений. Ч. 2 : [учебное пособие для 4 курса ФПИИ] / В. Э. Малышкин ; Новосиб. гос. техн. ун-т. - Новосибирск, 2000. - 51 с. : ил.
10. Кормен Т. Алгоритмы: построение и анализ / Т. Кормен, Ч. Лейзерсон, Р. Ривест ; [пер. с англ. К. Белов и др. ; под ред. А. Шеня]. - М., 1999. - 955 с. : ил.

В электронном виде

1. Малышкин В. Э. Параллельное программирование мультимикомпьютеров : [учебник] / В. Э. Малышкин, В. Д. Корнеев. - Новосибирск, 2006. - 295 с. : ил.. - Режим доступа: <http://www.ciu.nstu.ru/fulltext/textbooks/2006/malyshkin.pdf>
2. Высокопроизводительные вычисления на WINDOWS-кластерах [Электронный ресурс] : информ.-аналитический портал. – Режим доступа: <http://parallel.ru/>. – Загл. с экрана.
3. Dongarra J. Applied Parallel Computing [electronic resource] / : State of the Art in Scientific Computing / edited by Jack Dongarra, Kaj Madsen, Jerzy Wasniewski. - Berlin Heidelberg :, 2006 : v.: digital // Springer e-books. - Режим доступа: <http://dx.doi.org/10.1007/11558958>
4. Cappello F. Recent Advances in Parallel Virtual Machine and Message Passing Interface [electronic resource] : : 14th European PVM/MPI User's Group Meeting, Paris, France, September 30 - October 3, 2007. Proceedings // edited by Franck Cappello, Thomas Herault, Ja. - Berlin, Heidelberg :, 2007. : v.: digital // Springer e-books. - Режим доступа: <http://dx.doi.org/10.1007/978-3-540-75416-9>
5. Ropo M. Recent Advances in Parallel Virtual Machine and Message Passing Interface [electronic resource] : : 16th European PVM/MPI Users' Group Meeting, Espoo, Finland, September 7-10, 2009. Proceedings // edited by Matti Ropo, Jan Westerholm, Jack Dongarra. - Berlin, Heidelberg :,

8. Методическое и программное обеспечение

8.1 Методическое обеспечение

В электронном виде

1. Корнеев В. Д. Методические указания для лабораторных по курсу "Основы параллельного программирования" [Электронный ресурс] : учебно-методическое пособие / В. Д. Корнеев ; Новосиб. гос. техн. ун-т. - Новосибирск, [2011]. - Режим доступа: http://ciu.nstu.ru/fulltext/unofficial/2011/lib_874_1321433132.doc. - Загл. с экрана.

8.2 Программное обеспечение

2. Argonne National Laboratory , MPICH-2.0, Реализация стандартов MPI для разработки параллельных программ

3. Free Software Foundation, Ink , GCC, Набор компиляторов для разработки программ на языках C, C++, Фортран, включая средства параллельного программирования OpenMP

1. Red Hat Linux, Fedora Linux, серверная операционная система

9. Контролирующие материалы для аттестации студентов по дисциплине

Экзаменационные вопросы

1. Представление алгоритма. Определение реализации алгоритма, последовательного и параллельного представления алгоритма. Определение алгоритма (по Мальцеву).
2. Определение сети Петри. Разметка сети Петри. Правила срабатывания переходов. Работа сети.
3. Граф достижимости сети Петри. Понятия ограниченной и безопасной сети. Понятие тупиковой разметки сети Петри.
4. Определение процесса. Два главных типа взаимодействия параллельных процессов. Задача взаимного исключения (определение).
5. Понятие критического интервала, разделяемого и неразделяемого ресурса. Пример сети Петри, моделирующей взаимное исключение.
6. Семафоры (определение). Операции над семафорами. Пример сети Петри, моделирующей операции над семафорами.
7. Определение взаимной блокировки (дедлока). Необходимые условия дедлока. Привести пример сети Петри, допускающей дедлок.
8. Определение взаимной блокировки (дедлока). Два подхода к борьбе с дедлоками.
9. Механизм "условных критических интервалов". Пример решения задачи "читатели-писатели" с помощью этого механизма.
10. Монитор. Общее представление. Пример монитора для задачи "производитель-потребитель".
11. Модели параллельно-последовательного программирования. MPMD и SPMD модели программирования.
12. Параллельная программа разделения множеств (Дейкстры), идея доказательства её корректности.
13. Ускорение и эффективность вычислений. Закон Амдалла.
14. Событийное управление (определение). Типичные "локальные" ситуации, которые могут возникнуть в событийном управлении, представить сетью Петри.
15. Событийное управление. Операции над сетями Петри (присоединения, исключения, итераций, наложения, разметки), продемонстрировать примерами.
16. Потокное управление (определение). Операции: преобразователь, синхронизатор, распределитель, селектор, арбитр. Пример реализации этими операциями условного выражения: $\text{if } a < b \text{ then } a + c \text{ else } a - c$.
17. Потокное управление (определение). Волновые вычисления. Пример волнового процессора умножения матрицы на матрицу.
18. Динамическое управление (определение). Понятие программы в асинхронном динамическом программировании.
19. Вычислительная модель Э.Дейкстры (охраняемые команды). Вычислительная модель Ч.Хоара последовательных сообщающихся процессов.
20. Синхронные вычисления. Определение систолического вычислителя. Три фазы систолического алгоритма. Пример систолического процессора для умножения матрицы на вектор (привести схему и программу).

Образец экзаменационного билета

Министерство
образования РФ

НОВОСИБИРСКИЙ
ГОСУДАРСТВЕННЫЙ
ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ

Экзаменационный билет № 12

По дисциплине Основы параллельного программирования

Факультет ФПМИ

Курс 4

1. Модели параллельно-последовательного программирования. MPMD и SPMD модели программирования. Необходимость явного распараллеливания.
2. Определение процесса. Два главных типа взаимодействия параллельных процессов. Задача взаимного исключения (определение). Понятие критического интервала, разделяемого и неразделяемого ресурса. Пример сети Петри, моделирующей взаимное исключение.
3. Задача.

Составил

Корнеев В.Д.

Дата

г.

Задача

к экзаменационному билету № 12