

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Рябчин Владимир Васильевич
Должность: Директор
Дата подписания: 22.02.2023 09:10:10
Уникальный программный ключ:
937d0b737ee35db03895d495a275a8aac5224805

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«Национальный исследовательский ядерный университет «МИФИ»
Технологический институт –
филиал федерального государственного автономного образовательного учреждения высшего
образования «Национальный исследовательский ядерный университет «МИФИ»
(ТИ НИЯУ МИФИ)

**КАФЕДРА ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И ПРИКЛАДНОЙ
МАТЕМАТИКИ**

ОДОБРЕНО
Ученым советом ТИ НИЯУ МИФИ
Протокол № 5 от 31.08.2021 г.

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ

Параллельное программирование

(наименование дисциплины (модуля))

Направление **09.03.01 Информатика и вычислительная техника**
подготовки
Профиль подготовки **Системы автоматизированного проектирования в
машиностроении**
Квалификация (степень) выпускника **бакалавр**
Форма обучения **очная**

Семестр	7	Итого
Трудоемкость, кред.	2	2
Общий объем курса, час.	72	72
Лекции, час.	16	16
Практич. занятия, час.	16	16
Лаборат. работы, час.	16	16
В форме практической подготовки, час.	32	32
СРС, час.	24	24
КСР, час.	-	-
Форма контроля – зачет	-	-

г. Лесной – 2021 г.

АННОТАЦИЯ

В ходе освоения дисциплины «Параллельное программирование» студенты знакомятся с архитектурой современных многопроцессорных вычислительных систем, параллельной обработки информации, технологий организации параллельных вычислений на многопроцессорных вычислительных комплексах с распределенной или общей оперативной памятью.

1. ЦЕЛЬ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ

Целью учебной дисциплины «Параллельное программирование» является формирование у студентов теоретических знаний и практических навыков программирования параллельных и распределенных систем.

Главной **задачей** дисциплины является формирование у студентов теоретических знаний и практических навыков разработки алгоритмов и программ и их реализации на многопроцессорных вычислительных системах.

Учебные задачи дисциплины:

- Изучение типов распределенных вычислительных систем;
- Изучение основных архитектур параллельных вычислительных комплексов;
- Изучение теоретических основ параллельных вычислений, параллельных алгоритмов;
- Изучение основных технологий параллельных и распределенных вычислений;
- Изучение современных методов и средств, используемых при эксплуатации распределенных вычислительных систем;
- Изучение тенденций развития методов и средств организации распределенных вычислений.

2. МЕСТО УЧЕБНОЙ ДИСЦИПЛИНЫ В СТРУКТУРЕ ООП ВО

Дисциплина «Параллельное программирование» изучается студентами четвертого курса, входит в теоретический блок профессионального модуля раздела Б.1, вариативной части учебного плана по направлению подготовки «Информатика и вычислительная техника» профиля подготовки «Системы автоматизированного проектирования в машиностроении».

Дисциплина основывается на знаниях, полученных в результате освоения дисциплин «Введение в программирование», «Алгоритмизация и программирование», «Объектно-ориентированное программирование».

Изучение дисциплины необходимо для прохождения преддипломной практики, а также практической работы выпускников по специальности.

Указанные связи и содержание дисциплины «Параллельное программирование» дают обучающемуся системное представление о комплексе изучаемых дисциплин в соответствии ОС ВО НИЯУ МИФИ, что обеспечивает соответственный теоретический уровень и практическую направленность в системе обучения будущей деятельности бакалавра.

3. ФОРМИРУЕМЫЕ КОМПЕТЕНЦИИ И ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ

Процесс изучения дисциплины «Параллельное программирование» направлен на формирование следующих компетенций: ОПК-8; ОПК-9; ПК-3; ПК-7.2

Код компетенции	Компетенция
ОПК-8	Способен разрабатывать алгоритмы и программы, пригодные для практического применения
ОПК-9	Способен осваивать методики использования программных средств для решения практических задач
ПК-3	Способен разрабатывать модели и компоненты аппаратно-программных комплексов и баз данных, используя современные инструментальные средства и технологии
ПК-7.2	Способен проектировать, внедрять и сопровождать программное обеспечение, производить разработку ПО с использованием современных технологий и средств разработки, выбирать и обосновывать выбор методологии разработки ПО и язык программирования, проектировать реляционные базы данных

Индикаторами достижения компетенций являются:

Код компетенции	Код индикатора	Индикатор
ОПК-8	З-ОПК-8	Знать: алгоритмические языки программирования, операционные системы и оболочки, современные среды разработки программного обеспечения
	У-ОПК-8	Уметь: составлять алгоритмы, писать и отлаживать коды на языке программирования, тестировать работоспособность программы, интегрировать программные модули
	В-ОПК-8	Владеть: языком программирования; навыками отладки и тестирования работоспособности программы
ОПК-9	З-ОПК-9	Знать: классификацию программных средств и возможности их применения для решения практических задач
	У-ОПК-9	Уметь: находить и анализировать техническую документацию по использованию программного средства, выбирать и использовать необходимые функции программных средств для решения конкретной задачи
	В-ОПК-9	Владеть: способами описания методики использования программного средства для решения конкретной задачи в виде документа, презентации или видеоролика
ПК-3	З-ПК-3	Знать: схемотехнику логических схем, цифровых и запоминающих устройств, принципы построения и элементы микропроцессоров и микроконтроллеров, принципы работы программируемых логических матриц и программируемой матричной логики, основы объектно-ориентированного подхода к программированию, базы данных и системы управления базами данных для информационных систем различного назначения, принципы построения современных операционных систем и особенности их применения
	У-ПК-3	Уметь: строить логические схемы счетчиков, регистров, сумматоров и запоминающих устройств, строить временные диаграммы работы интерфейсов и контроллеров, сопрягать аппаратные и программные средства в составе аппаратно-программных комплексов, работать с современными системами программирования, включая объектно-ориентированные
	В-ПК-3	Владеть: современными инструментальными средствами проектирования цифровых устройств, языками процедурного и

Код компетенции	Код индикатора	Индикатор
		объектно-ориентированного программирования, навыками разработки и отладки программ
ПК-7.2	З-ПК-7.2	Знать: основные методологии разработки ПО, теорию проектирования реляционных баз данных, основные модели жизненного цикла ПО, способы тестирования ПО, основные подходы к процессу разработки ПО
	У-ПК-7.2	Уметь: выбирать методологию разработки ПО в зависимости от поставленной задачи, проектировать реляционные базы данных, выбирать наиболее подходящую модель жизненного цикла ПО, тестировать разработанное ПО
	В-ПК-7.2	Владеть: навыками проектирования реляционных БД, методами и приемами тестирования ПО, навыками командной разработки ПО, навыками использования различных технологий и средств разработки ПО

4. ВОСПИТАТЕЛЬНАЯ РАБОТА

Код	Направление/цели	Создание условий, обеспечивающих:	Использование воспитательного потенциала учебных дисциплин
B22	Профессиональное воспитание	формирование творческого инженерного мышления, навыков организации коллективной проектной деятельности	Использование воспитательного потенциала дисциплин профессионального модуля для: - формирования производственного коллективизма в ходе совместного решения как модельных, так и практических задач, а также путем подкрепление рационально-технологических навыков взаимодействия в проектной деятельности эмоциональным эффектом успешного взаимодействия, ощущением роста общей эффективности при распределении проектных задач в соответствии с сильными компетентностными и эмоциональными свойствами членов проектной группы.
B26		формирование ответственного и критического отношения к информации и информационным ресурсам	Использование воспитательного потенциала дисциплин профессионального модуля для формирования культуры решения изобретательских задач, развития логического мышления, путем вовлечения

Код	Направление/цели	Создание условий, обеспечивающих:	Использование воспитательного потенциала учебных дисциплин
			студентов в решение различных задач профессиональной деятельности.

Организация интерактивных мероприятий и реализация специализированных заданий с воспитательным и социальным акцентом:

- лабораторные занятия – совместная деятельность студентов, которая дает всем обучающимся возможность участвовать в работе, практиковать навыки сотрудничества, межличностного общения (в частности, умение активно слушать, вырабатывать общее мнение, разрешать возникающие разногласия).

Перечисленные мероприятия направлены на:

- формирование культуры решения изобретательских задач;
- формирование представления об основных задачах профессиональной деятельности;
- развитие творческого мышления для решения базовых задач;
- развитие способности работать в группе и коллективно решать поставленные задачи.

5. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

Общая трудоемкость дисциплины составляет 2 зачетных единицы, 72 часа.

№ п/п	Наименование раздела учебной дисциплины	Недели	Виды учебной деятельности, включая самостоятельную работу студентов и трудоемкость в ак. часах				Обязат. текущий контроль успеваемости (форма, неделя) ¹	Аттестация раздела (форма, неделя)	Максимальный балл за раздел
			Лекции	Практические работы	Лабораторные работы	Самостоятельные работы			
1	Раздел 1. Моделирование и анализ параллельных алгоритмов. Этапы и средства разработки параллельных программ.	1-8	8	10	10	14	ЛР1-4(8 б.) Зд (7 нед. – 8 б.)	КИ1	40

¹ ЛР – лабораторная работа, Зд – задание, Т – тест.

2	Раздел 2. Интерфейс передачи сообщений MPI. Технологии параллельного программирования и численные алгоритмы.	9-16	8	6	6	10	ЛР5-8(8 б.) Т (16 нед. – 8 б.)	КИ2	40
	Зачет								20
	ИТОГО:		16	16	16	24			100

НАИМЕНОВАНИЕ ТЕМ И СОДЕРЖАНИЕ ЛЕКЦИОННЫХ ЗАНЯТИЙ

Раздел 1. Моделирование и анализ параллельных алгоритмов. Этапы и средства разработки параллельных программ.

1. Состояние и проблемы параллельных вычислений.

Ограничение максимальной производительности однопроцессорных ЭВМ. Необходимость коллективного режима решения задач. Автоматизация управления распределенных технических систем. Технические требования по снижению стоимости и повышению надежности.

История введения параллелизма. Различие многозадачных, параллельных и распределенных вычислений. Проблемы использования параллелизма (закон Амдаля, закон Мура, гипотеза Минского, закон Гроша). Зависимость эффективности параллельных вычислений от учета особенностей аппаратуры. Сложность разработки параллельных алгоритмов. Трудоемкость проверки правильности параллельных программ.

2. Моделирование и анализ параллельных алгоритмов. Показатели качества

Компьютер с неограниченным параллелизмом (паракомпьютер). Модели многопроцессорных систем с общей и распределенной памятью. Модель конвейерной системы.

Модель алгоритма в виде графа "операнд - операция". Представление алгоритма в виде графа потока данных. Расписание параллельных вычислений. Показатель временной сложности алгоритма. Оценка времени выполнения алгоритма для паракомпьютера (предельное распараллеливание) и для систем с конечным количеством процессоров. Зависимость оценок от топологии графа алгоритма и необходимость оптимизации структуры графа. Способы получения оптимального расписания вычислений.

Модель параллельных вычислений в виде сети Петри. Основные понятия теории сетей Петри. Использование сетей Петри для описания параллельных вычислений. Демонстрация основных проблем параллельных вычислений: синхронизация, взаимоисключение, блокировка (тупики).

Модель параллельных вычислений в виде графа "процесс-ресурс". Понятие процесса. Синхронизация параллельных процессов. Аппарат событий.

Взаимоисключение параллельных процессов. Концепция ресурса. Механизмы взаимоисключения: алгоритм Деккера, семафоры (Дейкстра), мониторы (Вирт). Примеры решения стандартных задач взаимоисключения: кольцевой буфер, проблема "читатели и писатели".

Взаимодействие параллельных процессов посредством механизма передачи сообщений. Механизмы передачи: очереди, почтовые ящики, порты. Принцип рандеву в языках Ада и ОККАМ.

Проблемы взаимодействия процессов. Понятие тупика и условия его возникновения. Предотвращение тупиков. Алгоритм банкира. Обнаружение тупиков и восстановление состояния процессов.

Многозадачный режим работы ЭВМ как частный случай параллельной обработки.

3. Этапы разработки параллельных методов. Средства разработки параллельных программ

Оценка эффективности параллельных вычислений. Показатель эффекта распараллеливания (ускорение). Эффективность использования вычислительной системы. Способы оценки показателей. Основные характеристики вычислительной системы, влияющие на величины ускорения и эффективности (архитектура, количество процессоров, топология каналов передачи данных).

Оценка коммуникационной трудоемкости параллельных алгоритмов. Характеристики топологий сети передачи данных. Алгоритмы маршрутизации. Методы передачи данных.

Анализ трудоемкости основных операций передачи данных. Передача данных между двумя процессорами сети. Одиночная и множественная рассылка сообщений. Операция циклического сдвига.

Методы логического представления топологии коммуникационной среды. Отображение кольцевой топологии и топологии решетки на гиперкуб.

Уровни распараллеливания вычислений. Распараллеливание вычислений на уровне команд, выражений, программных модулей, отдельно выполняемых заданий.

Этапы построения параллельных алгоритмов и программ. Выбор параллельного алгоритма. Реализация алгоритма в виде параллельной программы. Построение исполняемой программы для параллельной вычислительной системы. Параллельное исполнение машинной программы. Частные постановки: выбор оптимального алгоритма для конкретной вычислительной системы, нахождение наилучшей топологии вычислительной системы для решения определенной задачи, распараллеливание существующего алгоритма.

Технологические аспекты распараллеливания. Декомпозиция алгоритма на параллельно исполняемые фрагменты вычислений. Распределение заданий по процессорам и балансировка. Синхронизация и взаимоисключение. Организация взаимодействия.

4. Интерфейс передачи сообщений MPI

Система MPI. Общая характеристика. Поддержка модели взаимодействия параллельных вычислителей при помощи передачи сообщений. Основные программные примитивы системы MPI.

Раздел 2. Интерфейс передачи сообщений MPI. Технологии параллельного программирования и численные алгоритмы.

1. Технология программирования OpenMP.

Общая характеристика стандарта OpenMP. Создание параллельных областей. Разделение вычислительной нагрузки между потоками. Работа с данными. Синхронизация. Функции и переменные окружения. Сравнительная характеристика подходов параллельного программирования для систем с распределенной и общей памятью.

2. Технология параллельного программирования CUDA.

Технология CUDA. Концепция параллельной виртуальной вычислительной машины и ее представление в виде распределенной неоднородной системы компьютеров.

Представление программной системы на виртуальной машине. Основные программные примитивы системы. Пример использования.

3. Параллельные численные алгоритмы для решения типовых задач вычислительной математики.

Общие способы распараллеливания алгоритмов. Выявление функциональной независимости отдельных фрагментов алгоритма (параллелизм команд). Геометрическое разделение вычислений (параллелизм данных). Иерархическая декомпозиция обработки данных.

Организация параллельного исполнения рекурсивных вычислений. Проблема рекурсивной зависимости этапов обработки данных. Каскадная схема. Подход для получения асимптотически ненулевой эффективности. Метод Оутса. Пример для вычисления частичных и общей сумм.

Параллельные численные алгоритмы линейной алгебры. Способы разбиения матриц (горизонтальная, вертикальная, блочные схемы). Методы вычисления произведения матриц с использованием разных схем разбиения матриц. Обеспечение предельно допустимого параллелизма. Обращение матриц. Параллельные методы решения систем линейных уравнений.

Параллельные численные алгоритмы решения дифференциальных уравнений в частных производных. Параллельная реализация прямых и итерационных методов решения дифференциальных уравнений в частных производных. Анализ разностных схем для эффективного разделения области определения решаемых задач.

Параллельные численные алгоритмы многомерной многоэкстремальной оптимизации. Характеристическая схема представления методов глобального поиска. Общий принцип распараллеливания методов. Оценка эффективности введения параллелизма: эффективность и безызбыточность. Синхронные и асинхронные варианты алгоритмов. Определение наилучших топологий вычислительной системы для реализации методов.

Аудиторные занятия и бюджет времени на самостоятельную подготовку студента

Учебная неделя	Наименование раздела, краткое наименование темы	Аудиторные занятия (час.)			Практическая подготовка	Самостоятель ная работа
		Лекции	Практическ ие работы	Лабораторн ые работы		
1	Моделирование и анализ параллельных алгоритмов. Этапы и средства разработки параллельных программ. Состояние и проблемы параллельных вычислений	2	2	2	4	3
3	Моделирование и анализ параллельных алгоритмов. Этапы и средства разработки параллельных программ. Моделирование и анализ параллельных алгоритмов. Показатели качества	2	2	2	4	4
5	Моделирование и анализ параллельных алгоритмов. Этапы и средства разработки параллельных программ. Этапы разработки параллельных методов. Средства разработки параллельных программ	2	4	2	6	4
7	Моделирование и анализ параллельных алгоритмов. Этапы и средства разработки параллельных программ. Интерфейс передачи сообщений MPI	2	2	4	6	3

Учебная неделя	Наименование раздела, краткое наименование темы	Аудиторные занятия (час.)			Практическая подготовка	Самостоятель ная работа
		Лекции	Практическ ие работы	Лабораторн ые работы		
9	Интерфейс передачи сообщений MPI. Технологии параллельного программирования и численные алгоритмы. Технология программирования OpenMP	2	2	2	4	3
11	Интерфейс передачи сообщений MPI. Технологии параллельного программирования и численные алгоритмы. Технология параллельного программирования CUDA	2	2	2	4	3
13	Интерфейс передачи сообщений MPI. Технологии параллельного программирования и численные алгоритмы. Параллельные численные алгоритмы для решения типовых задач вычислительной математики	4	2	2	4	4
	Итого	16	16	16	32	24

6. ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ

При проведении занятий и организации самостоятельной работы студентов используются:

1. Традиционные технологии обучения, предполагающие передачу информации в готовом виде, формирование учебных умений по образцу: лекция-изложение, лабораторные работы.

Использование традиционных технологий обеспечивает ориентирование студента в потоке информации, связанной с различными подходами к вопросам, рассматриваемым в пределах дисциплины; самоопределение в выборе оптимального пути и способов личностно-профессионального развития; систематизацию знаний, полученных студентами в процессе аудиторной и самостоятельной работы. Практические занятия обеспечивают развитие и закрепление умений и навыков.

2. Интерактивные технологии обучения, предполагающие организацию обучения как продуктивной творческой деятельности в режиме взаимодействия студентов друг с другом и с преподавателем. Использование интерактивных образовательных технологий способствует повышению интереса и мотивации учащихся, активизации мыслительной деятельности и творческого потенциала студентов, делает более эффективным усвоение материала, позволяет индивидуализировать обучение и ввести экстренную коррекцию знаний.

Организация самостоятельной работы студентов

Цель самостоятельной работы: закрепление знаний, полученных на занятиях. На самостоятельную работу по каждой теме выносятся следующие задания:

Задание 1. Выполнение индивидуального задания на зачет.

Задание 2. Составление глоссария по теме лекции.

7. ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ВХОДНОГО, ТЕКУЩЕГО КОНТРОЛЯ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

Фонд оценочных средств по дисциплине обеспечивает проверку освоения планируемых результатов обучения (компетенций и их индикаторов) посредством мероприятий текущего, рубежного и промежуточного контроля по дисциплине.

Связь между формируемыми компетенциями и формами контроля их освоения представлена в следующей таблице:

Код	Проектируемые результаты освоения дисциплины и индикаторы формирования компетенций			Средства и технологии оценки
	Знать (З)	Уметь (У)	Владеть (В)	
ОПК-8	З-ОПК-8	У-ОПК-8	В-ОПК-8	ЛР1-8, Зд, Т, З
ОПК-9	З-ОПК-9	У-ОПК-9	В-ОПК-9	ЛР1-8, Зд, Т, З
ПК-3	З-ПК-3	У-ПК-3	В-ПК-3	ЛР1-8, Зд, Т, З
ПК-7.2	З-ПК-7.2	У-ПК-7.2	В-ПК-7.2	ЛР1-8, Зд, Т, З

Шкала оценки за текущую аттестацию

Раздел	Форма текущего контроля	Максимальный балл	Максимальный балл за раздел
Раздел 1. Моделирование и анализ параллельных алгоритмов. Этапы и средства разработки параллельных программ			40
Лабораторные работы	ЛР1-4	8	
Задание	Зд	8	
Раздел 2. Интерфейс передачи сообщений MPI. Технологии параллельного программирования и численные алгоритмы			40
Лабораторные работы	ЛР5-8	8	
Тест	Т	8	
Итого			80

Шкала оценки за промежуточную аттестацию (зачет)

Критерий оценивания	Шкала оценивания
студент полностью раскрыл содержание теоретических вопросов, самостоятельно, без наводящих вопросов, решил предложенную задачу, объяснил и мотивировал решение задачи, смог разъяснить особенности применения теоретических знаний на практике, что может выражаться в уверенных ответах на дополнительные вопросы преподавателя.	20-18
студент раскрыл содержание теоретических вопросов, продемонстрировал знания основных понятий и определений, знание специфических для рассматриваемого раздела терминов и их понимание, что может выражаться в уверенном ответе на вопросы преподавателя, но не смог сразу разъяснить особенности применения теоретических знаний на практике.	17-15

Критерий оценивания	Шкала оценивания
студент раскрыл содержание вопросов с большими затруднениями, требовалась помощь преподавателями в форме наводящих вопросов, напоминания алгоритмов решения задачи, студент затруднялся в объяснении решения задачи	14-12
студент не смог раскрыть содержание теоретических вопросов, продемонстрировать знания в решении задачи, даже если преподаватель пытался помочь в форме наводящих вопросов и напоминания алгоритмов решения задачи	11-0

Шкала итоговой оценки за семестр

Итоговая оценка представляет собой сумму баллов, заработанных студентом при выполнении заданий в рамках текущего и промежуточного контроля и выставляется в соответствии с Положением о кредитно-модульной системе в соответствии со следующей шкалой:

Оценка по 4-балльной шкале	Сумма баллов	Оценка ECTS
5 – «отлично»	90-100	A
4 – «хорошо»	85-89	B
	75-84	C
	70-74	D
3 – «удовлетворительно»	65-69	E
	60-64	F
2 – «неудовлетворительно»	Ниже 60	F

Расшифровка уровня знаний, соответствующего полученным баллам, дается в таблице указанной ниже

Сумма баллов	Оценка ECTS	Уровень приобретенных знаний по дисциплине
90-100	A	«Отлично» - теоретическое содержание курса освоено полностью, без пробелов, необходимые практические навыки работы с освоенным материалом сформированы, все предусмотренные программой обучения учебные задания выполнены, качество их выполнения оценено числом баллов, близким к максимальному.
85-89	B	«Очень хорошо» - теоретическое содержание курса освоено полностью, без пробелов, необходимые практические навыки работы с освоенным материалом в основном сформированы, все предусмотренные программой обучения учебные задания выполнены, качество выполнения большинства из них оценено числом баллов, близким к максимальному.
75-84	C	«Хорошо» - теоретическое содержание курса освоено полностью, без пробелов, некоторые практические навыки работы с освоенным материалом сформированы недостаточно, все предусмотренные программой обучения учебные задания выполнены, качество выполнения ни одного из них не оценено минимальным числом баллов, некоторые виды заданий выполнены с ошибками.
65-74	D	«Удовлетворительно» - теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые практические навыки работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий, возможно, содержат ошибки.

Сумма баллов	Оценка ECTS	Уровень приобретенных знаний по дисциплине
60-64	E	«Посредственно» - теоретическое содержание курса освоено частично, некоторые практические навыки работы не сформированы, многие предусмотренные программой обучения учебные задания не выполнены, либо качество выполнения некоторых из них оценено числом баллов, близким к минимальному.
Ниже 60	F	«Неудовлетворительно» - очень слабые знания, недостаточные для понимания курса, имеется большое количество основных ошибок и недочетов.

Студент считается аттестованным по разделу, экзамену, если он набрал не менее 60% от максимального балла, предусмотренного рабочей программой.

Контрольные мероприятия, за которые студент получил 0 баллов (неявка в установленный срок), подлежат обязательной пересдаче. Сроки пересдач контрольных мероприятий в течение семестра определяет кафедра.

Студент, пропустивший контрольное мероприятие без уважительной причины или получивший за него неудовлетворительную оценку, после пересдачи контрольного мероприятия получает балл ниже установленного на 25%.

Вопросы к зачету по дисциплине «Параллельное программирование»

- 1) Ограничение максимальной производительности однопроцессорных ЭВМ. Необходимость коллективного режима решения задач.
- 2) Автоматизация управления распределенных технических систем. Технические требования по снижению стоимости и повышению надежности.
- 3) История введения параллелизма. Различие многозадачных, параллельных и распределенных вычислений. Проблемы использования параллелизма (закон Амдаля, закон Мура, гипотеза Минского, закон Гроша).
- 4) Зависимость эффективности параллельных вычислений от учета особенностей аппаратуры. Сложность разработки параллельных алгоритмов.
- 5) Трудоемкость проверки правильности параллельных программ.
- 6) Компьютер с неограниченным параллелизмом (паракомпьютер). Модели многопроцессорных систем с общей и распределенной памятью. Модель конвейерной системы.
- 7) Модель алгоритма в виде графа "операнд - операции". Представление алгоритма в виде графа потока данных. Расписание параллельных вычислений. Показатель временной сложности алгоритма.
- 8) Оценка времени выполнения алгоритма для паракомпьютера (предельное распараллеливание) и для систем с конечным количеством процессоров.
- 9) Зависимость оценок от топологии графа алгоритма и необходимость оптимизации структуры графа. Способы получения оптимального расписания вычислений.
- 10) Модель параллельных вычислений в виде сети Петри. Основные понятия теории сетей Петри. Использование сетей Петри для описания параллельных вычислений.
- 11) Демонстрация основных проблем параллельных вычислений: синхронизация, взаимное исключение, блокировка (тупики).
- 12) Модель параллельных вычислений в виде графа "процесс-ресурс". Понятие процесса. Синхронизация параллельных процессов. Аппарат событий.
- 13) Взаимное исключение параллельных процессов. Концепция ресурса. Механизмы взаимного исключения: алгоритм Деккера, семафоры (Дейкстра), мониторы (Вирт).

- 14) Взаимодействие параллельных процессов посредством механизма передачи сообщений. Механизмы передачи: очереди, почтовые ящики, порты. Принцип рандеву в языках Ада и ОККАМ.
- 15) Проблемы взаимодействия процессов. Понятие тупика и условия его возникновения. Предотвращение тупиков. Алгоритм банкира. Обнаружение тупиков и восстановление состояния процессов.
- 16) Многозадачный режим работы ЭВМ как частный случай параллельной обработки.
- 17) Оценка эффективности параллельных вычислений. Показатель эффекта распараллеливания (ускорение). Эффективность использования вычислительной системы. Способы оценки показателей.
- 18) Основные характеристики вычислительной системы, влияющие на величины ускорения и эффективности (архитектура, количество процессоров, топология каналов передачи данных).
- 19) Оценка коммуникационной трудоемкости параллельных алгоритмов. Характеристики топологий сети передачи данных. Алгоритмы маршрутизации. Методы передачи данных.
- 20) Анализ трудоемкости основных операций передачи данных. Передача данных между двумя процессорами сети. Одиночная и множественная рассылка сообщений. Операция циклического сдвига.
- 21) Методы логического представления топологии коммуникационной среды. Отображение кольцевой топологии и топологии решетки на гиперкуб.
- 22) Уровни распараллеливания вычислений. Распараллеливание вычислений на уровне команд, выражений, программных модулей, отдельно выполняемых заданий.
- 23) Этапы построения параллельных алгоритмов и программ. Выбор параллельного алгоритма. Реализация алгоритма в виде параллельной программы. Построение исполняемой программы для параллельной вычислительной системы.
- 24) Параллельное исполнение машинной программы. Частные постановки: выбор оптимального алгоритма для конкретной вычислительной системы, нахождение наилучшей топологии вычислительной системы для решения определенной задачи, распараллеливание существующего алгоритма.
- 25) Технологические аспекты распараллеливания. Декомпозиция алгоритма на параллельно исполняемые фрагменты вычислений. Распределение заданий по процессорам и балансировка. Синхронизация и взаимоисключение. Организация взаимодействия.
- 26) Система MPI. Общая характеристика. Поддержка модели взаимодействия параллельных вычислителей при помощи передачи сообщений. Основные программные примитивы системы MPI.
- 27) Общая характеристика стандарта OpenMP. Создание параллельных областей. Разделение вычислительной нагрузки между потоками. Работа с данными. Синхронизация. Функции и переменные окружения.
- 28) Сравнительная характеристика подходов параллельного программирования для систем с распределенной и общей памятью.
- 29) Технология CUDA. Концепция параллельной виртуальной вычислительной машины и ее представление в виде распределенной неоднородной системы компьютеров.
- 30) Представление программной системы на виртуальной машине. Основные программные примитивы системы. Пример использования.
- 31) Общие способы распараллеливания алгоритмов. Выявление функциональной независимости отдельных фрагментов алгоритма (параллелизм команд).
- 32) Геометрическое разделение вычислений (параллелизм данных). Иерархическая декомпозиция обработки данных.

- 33) Организация параллельного исполнения рекурсивных вычислений. Проблема рекурсивной зависимости этапов обработки данных.
- 34) Каскадная схема. Подход для получения асимптотически ненулевой эффективности. Метод Оутса. Пример для вычисления частичных и общей сумм.
- 35) Параллельные численные алгоритмы линейной алгебры. Способы разбиения матриц (горизонтальная, вертикальная, блочные схемы).
- 36) Методы вычисления произведения матриц с использованием разных схем разбиения матриц. Обеспечение предельно допустимого параллелизма. Обращение матриц. Параллельные методы решения систем линейных уравнений.
- 37) Параллельные численные алгоритмы решения дифференциальных уравнений в частных производных.
- 38) Параллельная реализация прямых и итерационных методов решения дифференциальных уравнений в частных производных. Анализ разностных схем для эффективного разделения области определения решаемых задач.
- 39) Параллельные численные алгоритмы многомерной многоэкстремальной оптимизации. Характеристическая схема представления методов глобального поиска. Общий принцип распараллеливания методов.
- 40) Оценка эффективности введения параллелизма: эффективность и безызбыточность. Синхронные и асинхронные варианты алгоритмов. Определение наилучших топологий вычислительной системы для реализации методов.

8. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

Основная литература

1. Федотов, И. Е. Параллельное программирование. Модели и приемы / И. Е. Федотов. — Москва : СОЛОН-ПРЕСС, 2018. — 390 с. — ISBN 978-5-91359-222-4. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <https://www.iprbookshop.ru/90420.html> (дата обращения: 12.11.2021). — Режим доступа: для авторизир. пользователей.

Дополнительная литература

1. Абрамян, М. Э. Практикум по параллельному программированию с использованием электронного задачника Programming Taskbook for MPI / М. Э. Абрамян. — Ростов-на-Дону : Издательство Южного федерального университета, 2010. — 172 с. — ISBN 978-5-9275-0778-8. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <https://www.iprbookshop.ru/47085.html> (дата обращения: 12.11.2021). — Режим доступа: для авторизир. пользователей.

2. Антонов, А. С. Параллельное программирование с использованием технологии MPI : учебное пособие / А. С. Антонов. — 3-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2021. — 83 с. — ISBN 978-5-4497-0934-9. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <https://www.iprbookshop.ru/102043.html> (дата обращения: 12.11.2021). — Режим доступа: для авторизир. пользователей.

3. Арыков, С. Б. Параллельное программирование над общей памятью. OpenMP : учебное пособие / С. Б. Арыков, М. А. Городничев, Г. А. Щукин. — Новосибирск : Новосибирский государственный технический университет, 2019. — 95 с. — ISBN 978-5-7782-3796-4. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <https://www.iprbookshop.ru/99203.html> (дата обращения: 12.11.2021). — Режим доступа: для авторизир. пользователей.

4. Биллиг, В. А. Параллельные вычисления и многопоточное программирование : учебник / В. А. Биллиг. — 3-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2021. — 310 с. — ISBN 978-5-4497-0936-3. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <https://www.iprbookshop.ru/102044.html> (дата обращения: 12.11.2021). — Режим доступа: для авторизир. пользователей.

Программное обеспечение:

1. MS Office
2. Visual Studio 2019

LMS и Интернет-ресурсы:

1. Образовательный портал НИЯУ МИФИ . URL: <https://online.mephi.ru/>
2. Центр информационно-библиотечного обеспечения учебно-научной деятельности НИЯУ МИФИ URL: <http://library.mephi.ru/>.
3. Электронная информационно-образовательная среда ТИ НИЯУ МИФИ URL: <http://stud.mephi3.ru/>.
4. Электронно-библиотечная система URL: IPRbooks <https://www.iprbookshop.ru/>.

9. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

Аудитория для проведения занятий лекционного типа, занятий семинарского типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, для самостоятельной работы:

проектор Nec + экран (настенный), компьютер: процессор IntelPentium 4; оперативная память 4GBDDR3; монитор ЖК Benq 19,5”, клавиатура, мышь, Adobe Reader

Для проведения лабораторных работ необходима компьютерная лаборатория, оснащенная рабочими местами для каждого студента, а так же рабочим местом преподавателя. Рабочее место оснащено компьютером: процессор IntelPentium 4; оперативная память 4GBDDR3; монитор ЖК Benq 19,5”, клавиатура, мышь.

Каждый студент имеет свой логин и пароль для входа в Электронную информационно-образовательную среду ТИ НИЯУ МИФИ (<http://stud.mephi3.ru/>)

Каждый студент имеет доступ к электронно-библиотечной системе IPRbooks.

Программа составлена в соответствии с требованиями ОС ВО НИЯУ МИФИ по направлению подготовки 09.03.01 «Информатика и вычислительная техника».

Автор: старший преподаватель кафедры «Информационных технологий и прикладной математики» Д.А. Ваганов