

МИНИСТЕРСТВО ТРАНСПОРТА РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«РОССИЙСКИЙ УНИВЕРСИТЕТ ТРАНСПОРТА»
(РУТ (МИИТ))



Рабочая программа дисциплины (модуля),
как компонент образовательной программы
высшего образования - программы магистратуры
по направлению подготовки
27.04.04 Управление в технических системах,
утвержденной первым проректором РУТ (МИИТ)
Тимониным В.С.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

Основы параллельного программирования

Направление подготовки: 27.04.04 Управление в технических системах

Направленность (профиль): Интеллектуальное управление в
транспортных системах

Форма обучения: Очная

Рабочая программа дисциплины (модуля) в виде
электронного документа выгружена из единой
корпоративной информационной системы управления
университетом и соответствует оригиналу

Простая электронная подпись, выданная РУТ (МИИТ)
ID подписи: 2053
Подписал: заведующий кафедрой Баранов Леонид Аврамович
Дата: 01.06.2025

1. Общие сведения о дисциплине (модуле).

Целями освоения учебной дисциплины (модуля) «Основы параллельного программирования» является овладение обучающимися навыков работы с современным программным обеспечением, в современных средах программирования, используемым при проектировании и разработке комплексных информационных систем управления, для которых недостаточно использование одного вычислительного потока или одного вычислительного компонента / блока, находящегося без увязки с другими вычислительными компонентами / блоками.

Задача: Формирование у обучающихся профессиональных компетенций, необходимых для проектирования, разработки и оптимизации параллельных алгоритмов и программ, функционирующих в распределенных и многопоточных средах, с целью создания масштабируемых и отказоустойчивых информационных систем управления, способных решать сложные задачи за счет эффективного использования множества вычислительных ресурсов.

2. Планируемые результаты обучения по дисциплине (модулю).

Перечень формируемых результатов освоения образовательной программы (компетенций) в результате обучения по дисциплине (модулю):

ПК-8 - Способен разрабатывать методическое, информационное, математическое, программное и аппаратное обеспечение автоматизированных средств обучения и повышения квалификации обучающихся;

ПК-11 - Способен к подготовке и осуществлению повышения квалификации кадров высшей квалификации, в том числе с использованием современных методов и технологий обучения;

ПК-17 - Способен анализировать национальный и международный опыта разработки и внедрения АСУП.

Обучение по дисциплине (модулю) предполагает, что по его результатам обучающийся будет:

Знать:

- Современное программное и аппаратное обеспечение в области параллельных вычислений (многоядерные процессоры, графические ускорители, кластерные системы), которое может быть использовано для создания учебных стендов и виртуальных лабораторий.

- Критерии и методики оценки эффективности программ повышения квалификации, включая сбор и анализ обратной связи от слушателей.

- Национальные и международные стандарты в области разработки программного обеспечения, качества ПО, информационной безопасности, применимые к созданию и внедрению АСУП

Уметь:

- Разрабатывать контрольно-измерительные материалы (тесты, задания для проверочных работ), позволяющие объективно оценить уровень сформированности компетенций в области параллельного программирования

- Адаптировать стиль и методы преподавания в зависимости от уровня подготовки и профессиональных интересов аудитории (например, для разработчиков приложений, системных программистов, научных сотрудников).

- Выявлять перспективные направления развития АСУП, связанные с использованием больших данных (Big Data) и технологий искусственного интеллекта, требующих применения параллельных вычислений

Владеть:

- Навыками использования систем контроля версий (например, Git) и сред автоматической проверки кода для организации учебного процесса и оценки заданий по параллельному программированию.

- Навыками публичного выступления и педагогического общения, позволяющими доходчиво объяснять сложные технические концепции.

- Навыками составления аналитических обзоров и отчетов по результатам анализа национального и международного опыта внедрения параллельных технологий в АСУП.

3. Объем дисциплины (модуля).

3.1. Общая трудоемкость дисциплины (модуля).

Общая трудоемкость дисциплины (модуля) составляет 7 з.е. (252 академических часа(ов)).

3.2. Объем дисциплины (модуля) в форме контактной работы обучающихся с педагогическими работниками и (или) лицами, привлекаемыми к реализации образовательной программы на иных условиях, при проведении учебных занятий:

Тип учебных занятий	Количество часов	
	Всего	Семестр №2
Контактная работа при проведении учебных занятий (всего):	48	48

В том числе:		
Занятия лекционного типа	32	32
Занятия семинарского типа	16	16

3.3. Объем дисциплины (модуля) в форме самостоятельной работы обучающихся, а также в форме контактной работы обучающихся с педагогическими работниками и (или) лицами, привлекаемыми к реализации образовательной программы на иных условиях, при проведении промежуточной аттестации составляет 204 академических часа (ов).

3.4. При обучении по индивидуальному учебному плану, в том числе при ускоренном обучении, объем дисциплины (модуля) может быть реализован полностью в форме самостоятельной работы обучающихся, а также в форме контактной работы обучающихся с педагогическими работниками и (или) лицами, привлекаемыми к реализации образовательной программы на иных условиях, при проведении промежуточной аттестации.

4. Содержание дисциплины (модуля).

4.1. Занятия лекционного типа.

№ п/п	Тематика лекционных занятий / краткое содержание
1	Введение в параллельное программирование Рассматриваемые вопросы: - основные понятия и термины параллельного программирования
2	Настройка сред программирования Рассматриваемы вопросы: - Настройка сред программирования под условия, необходимые для запуска параллельного кода
3	Параллельный код Рассматриваемые вопросы: - особенности тестирования и отладки параллельного кода - особенности параллельного кода
4	Обработка событий при параллельном программировании Рассматриваемые вопросы: - особенности работы параллельного программирования - обработка событий при параллельном программировании
5	Среды программирования, адаптированные под параллельные вычисления Рассматриваемые вопросы: - основные понятия и функции среды программирования - особенности среды программирования, адаптированной под параллельные вычисления
6	Структурирование и оптимизация параллельного кода Рассматриваемые вопросы: - структурирование параллельного кода - оптимизация параллельного кода

№ п/п	Тематика лекционных занятий / краткое содержание
7	Параллельные вычисления Рассматриваемые вопросы: - Чтение и запись баз данных с применением методов параллельных вычислений
8	Синхронизация и взаимодействие процессов Рассматриваемые вопросы: - особенности синхронизации и взаимодействия процессов
9	Конвейеризация Рассматриваемые вопросы: - основные понятия конвейеризации - особенности конвейеризации
10	Кэш-память Рассматриваемые вопросы: - основные понятия кэш-памяти - особенности функционирования кэш-памяти - программная реализация
11	Задачи параллельного программирования Рассматриваемые вопросы: - классические задачи параллельного программирования
12	Управление задачами Рассматриваемые вопросы: - Синхронизация и взаимодействие параллельных задач - Интерфейс передачи сообщений (MPI)

4.2. Занятия семинарского типа.

Лабораторные работы

№ п/п	Наименование лабораторных работ / краткое содержание
1	Настройка инструментальной среды и шаблонов параллельного кода В результате выполнения студент умеет формировать шаблоны ПО для параллельных вычислений, модернизировать условия программирования (настраивать профили сборки, линковщики, параметры компилятора), а также настраивать и исследовать встроенные отладчики для работы с многопоточными приложениями.
2	Анализ событий и поведения параллельных программ В результате студент умеет использовать обработчики событий (профилировщики, трейсеры) для анализа работы параллельного кода, исследовать его поведение в различных средах (Windows, Linux, различные IDE) и интерпретировать полученные данные для выявления узких мест.
3	Методы оптимизации параллельных вычислений на уровне архитектуры В результате студент умеет применять основные методы оптимизации (распараллеливание циклов, устранение ложного разделения кэша) и эффективно использовать иерархию кэш-памяти микропроцессора для минимизации задержек при доступе к данным.
4	Управление жизненным циклом параллельных задач В результате студент умеет создавать (запускать) потоки и задачи, корректно завершать (отменять) их, управлять временем их жизни и обрабатывать состояния, связанные с ожиданием и завершением.
5	Синхронизация и взаимодействие параллельных процессов В результате студент умеет организовывать взаимодействие между потоками/процессами,

№ п/п	Наименование лабораторных работ / краткое содержание
	настраивать их синхронизацию с помощью примитивов (мьютексы, семафоры, события) для решения задач, требующих согласованного доступа к ресурсам и обмена данными.
6	Реализация конвейерной обработки данных В результате работы студент умеет проектировать и реализовывать конвейерные вычислительные процессы, распределяя этапы обработки данных между параллельными исполнителями для увеличения пропускной способности системы.
7	Классические алгоритмы и работа с данными в параллельных системах В результате выполнения работы студент умеет решать классические задачи параллельного программирования (производители-потребители, читатели-писатели, обедающие философы), а также применять методы параллельных вычислений для эффективного чтения и записи в базы данных.
8	Применение MPI для распределенных вычислений В результате выполнения работы студент умеет использовать интерфейс передачи сообщений (MPI) для решения задач предметной области в распределенной среде (кластеры, вычислительные сети), организуя обмен данными между узлами и синхронизацию вычислений.

4.3. Самостоятельная работа обучающихся.

№ п/п	Вид самостоятельной работы
1	Изучение дополнительной литературы.
2	Подготовка к лабораторным работам.
3	Выполнение курсовой работы.
4	Подготовка к промежуточной аттестации.
5	Подготовка к текущему контролю.

4.4. Примерный перечень тем курсовых работ

1. Отработка механизма задержки отрисовки на примере метода градиентного спуска (любого-другого метода оптимизации). Показывать график функции (на плоскости). Показывать точками шаги алгоритма поиска по методу градиентного спуска с задержкой в три секунды между двумя отрисовываемыми точками.

2. Обрамление и форматирование ячеек в электронной таблице Microsoft Office Excel при использовании подключаемой библиотеки «Sheet.JS». Проверка и демонстрация возможностей библиотеки «Sheet.JS». Описание входящих в состав библиотеки методов. Инструкция пользователя по применению этих методов.

3. Составление электронных таблиц Microsoft Office Excel в обход использования подключаемых JavaScript библиотек, которые в свою очередь интегрируются в объектную модель Microsoft Office Excel. Создание текстовых файлов XML-структуры, размещённых в правильной структуре создаваемых каталогов и подкаталогов, упакованных в архив *.rar или *.zip,

которые необходимо переименовать в *.xlsx (вручную или автоматически) и запускать в Microsoft Office Excel.

4. Создание графика поверхности с возможностью вращения координатных осей и перерисовки поверхности в зависимости от выбранных углов поворота.

5. Создание графика поверхности с фиксированными осями, но с визуализацией изменения уровней (слоёв): уровни поверхности необходимо расцветчивать градиентно, в соответствии с линиями уровня.

6. Создание и настройка графика линий уровня для поверхностей, задаваемых различными функциональными зависимостями.

7. Размещение области векторной графики (SVG – Scalable Vector Graphics) в контейнере, занимающем фиксированную экранную площадь 800 x 600, измеряемую в пикселях. SVG занимает площадь по диапазону беззнакового типа Int16 (от 0 до 65535). Следует заполнить эту область прямыми, наклонёнными под 45 градусов к оси абсцисс и отстоящими друг от друга на 100 пикселей; наклонёнными под 135 градусов к оси абсцисс и так же отстоящими друг от друга на 100 пикселей.

8. Реализация механизма Drag & Drop для перемещения объектов, изображённых на графической области SVG. Например, изображаются треугольник и круг, их можно перетаскивать куда угодно внутри графической области. Необходимо предусмотреть запрет на перетаскивание фигур за пределы границ графической области.

9. Создание таблицы на базе компонента Grid (сетка) и предоставление возможности увеличения и уменьшения размеров строк и столбцов вводимыми численными уставками или курсором мыши.

10. Работа по анализу изображений и цифровой фильтрации изображений свёрточным алгоритмом.

11. Работа по построению гистограмм для загруженных в среду / в браузер изображений (форматы: *.jpeg, *.png, *.gif, *.bmp).

12. Реализация интерфейса для исследования фигур Лиссажу (ползунки, окна ввода уставок, логические признаки, масштабирование).

13. Настройка прямого или косвенного сопряжения «Vue.js» с базой данных Microsoft Office Access.

14. Настройка сопряжения «Vue.js» с документами Microsoft Office Word для чтения, редактирования и записи.

15. Подключаемые библиотеки JavaScript и возможности их практического применения в учебном процессе на кафедре (реферативно-аналитическая работа).

16. Реализация игры «Крестики-нолики» на базе фреймворка «Vue.js».

17. Реализация игры «Пятнашки» на базе фреймворка «Vue.js».
18. Реализация игры «Сапёр» на базе фреймворка «Vue.js».
19. Реализация сопряжения «Vue.js» с Microsoft Office Visio на примере составления блок-схемы алгоритма по анализируемому системой коду, считанному в браузер из текстового файла.

5. Перечень изданий, которые рекомендуется использовать при освоении дисциплины (модуля).

№ п/п	Библиографическое описание	Место доступа
1	Основы параллельного программирования Богачёв К. Ю. Издательство "Лаборатория знаний". - 5-е эл.изд. - 345 с. - ISBN 978-5-93208-802-9 , 2024	https://reader.lanbook.com/book/458324
2	Парадигма программирования Городня Л. В. Учебное пособие Издательство "Лань". - 232 с. , 2021	https://reader.lanbook.com/book/151660#227

6. Перечень современных профессиональных баз данных и информационных справочных систем, которые могут использоваться при освоении дисциплины (модуля).

Официальный сайт РУТ (МИИТ) (<https://www.miit.ru/>).

Научно-техническая библиотека РУТ (МИИТ) (<http://library.miit.ru>).

Образовательная платформа «Юрайт» (<https://urait.ru/>).

Общие информационные, справочные и поисковые системы «Консультант Плюс», «Гарант».

Электронно-библиотечная система издательства «Лань» (<http://e.lanbook.com/>).

Электронно-библиотечная система ibooks.ru (<http://ibooks.ru/>).

7. Перечень лицензионного и свободно распространяемого программного обеспечения, в том числе отечественного производства, необходимого для освоения дисциплины (модуля).

Microsoft Internet Explorer (или другой браузер).

Операционная система Microsoft Windows.

Microsoft Office.

Microsoft Visual Studio 2015.

Веб-браузер Google Chrome.

8. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю).

Учебные аудитории для проведения учебных занятий, оснащенные компьютерной техникой и наборами демонстрационного оборудования.

9. Форма промежуточной аттестации:

Курсовая работа во 2 семестре.

Экзамен во 2 семестре.

10. Оценочные материалы.

Оценочные материалы, применяемые при проведении промежуточной аттестации, разрабатываются в соответствии с локальным нормативным актом РУТ (МИИТ).

Авторы:

доцент, доцент, к.н. кафедры
«Управление и защита
информации»

А.И. Сафронов

Согласовано:

Заведующий кафедрой УиЗИ

Л.А. Баранов

Председатель учебно-методической
комиссии

С.В. Володин