

МИНИСТЕРСТВО ТРАНСПОРТА РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«РОССИЙСКИЙ УНИВЕРСИТЕТ ТРАНСПОРТА»

Кафедра «Цифровые технологии управления транспортными процессами»

АННОТАЦИЯ К РАБОЧЕЙ ПРОГРАММЕ ДИСЦИПЛИНЫ

«Асинхронное и параллельное программирование»

Направление подготовки:	09.03.01 – Информатика и вычислительная техника
Профиль:	Программное обеспечение средств вычислительной техники и автоматизированных систем
Квалификация выпускника:	Бакалавр
Форма обучения:	очная
Год начала подготовки	2020

1. Цели освоения учебной дисциплины

Целью дисциплины «Параллельное программирование» является формирование у студента систематизированных знаний об основных принципах построения и методах разработки параллельных алгоритмов и программ, освоение методов и технологий проектирования программного обеспечения для работы параллельных процессов. Основной целью изучения учебной дисциплины «Параллельное программирование» является формирование компетенций в области проектирования и отладки распределенного многопроцессорного (многопоточного) программного обеспечения (ПО)

2. Место учебной дисциплины в структуре ОП ВО

Учебная дисциплина "Асинхронное и параллельное программирование" относится к блоку 1 "Дисциплины (модули)" и входит в его вариативную часть.

3. Планируемые результаты обучения по дисциплине (модулю), соотнесенные с планируемыми результатами освоения образовательной программы

Процесс изучения дисциплины направлен на формирование следующих компетенций:

ПКР-1	Способность выполнять работы и управлять работами по разработке архитектур и прототипов информационных систем (ИС)
УК-2	Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм и имеющихся ресурсов и ограничений

4. Общая трудоемкость дисциплины составляет

4 зачетные единицы (144 ак. ч.).

5. Образовательные технологии

Преподавание дисциплины «Параллельное программирование» осуществляется в форме лекций, лабораторных работ, самостоятельной работы студентов. Лекции проводятся в форме мультимедиа-лекций, на которых демонстрируются презентации. Студенты имеют возможность ознакомиться с материалами презентации до начала лекции. Лабораторные работы проводятся в компьютерном классе с установленным программным обеспечением, необходимым для решения индивидуальных задач. На практических работах выполняются индивидуальные задания, демонстрируются готовые части выполненных заданий и отчета по заданию. Разработка проектов по индивидуальным заданиям ведется с использованием офисного пакета и интерактивных средств разработки на языке программирования C++/C#/Java. Самостоятельная работа студента организована с использованием традиционных видов работы и интерактивных технологий. Оценка полученных знаний, умений и навыков основана на модульно-рейтинговой технологии. Весь курс разбит на 3 раздела, представляющих собой логически заверченный объем учебной информации. Фонды оценочных средств освоенных компетенций включают как вопросы теоретического характера для оценки знаний, так и задания практического содержания (индивидуальные задания) для оценки умений и навыков. Теоретические знания проверяются путём решения тестов с использованием компьютеров и в ходе проверки отчетов по выполненным индивидуальным работам..

6. Содержание дисциплины (модуля), структурированное по темам (разделам)

РАЗДЕЛ 1

Раздел 1. Процессы и потоки.

Тема 1. Введение в проблематику параллельного программирования.

Тема 2. Процесс. Состояния процесса Переключение контекста.

Тема 3. Потоки, их создание и завершение. Переключение потоков.

РАЗДЕЛ 2

Раздел 2. Процесс разработки параллельной программы.

Тема 4. Основные проблемы при разработке параллельного алгоритма.

Тема 5. Стратегии размещения задач.

Тема 6. Модели решений при разработке параллельных алгоритмов.

РАЗДЕЛ 3

Раздел 3. Межпроцессное взаимодействие

Тема 7. Ресурсы и их характеристики.

Тема 8. Проблемы взаимодействия процессов.

Тема 9. Межпроцессное взаимодействие.

Тема 10. Примитивы синхронизации: критические секции. Алгоритм Петерсона.

Тема 11. Примитивы синхронизации: Семафоры.

Тема 12. Примитивы синхронизации: Мьютексы.

РАЗДЕЛ 4

Раздел 4. Решение классических задач.

Тема 13. Решение классической задачи межпроцессного взаимодействия («Обедающие философы», «Читатели и писатели», «Спящий парикмахер»)

Тема 14. Вычисление значения определенного интеграла по методу адаптивной квадратуры в условиях разделяемой памяти

Тема 15. Вычисление произведения матриц с использованием конвейерного алгоритма в условиях распределенной памяти

Экзамен