

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

СОГЛАСОВАНО:

Выпускающая кафедра УТБиИС
Доцент



В.Е. Нутович

27 апреля 2020 г.

УТВЕРЖДАЮ:

Директор ИУЦТ



С.П. Вакуленко

16 сентября 2020 г.



Кафедра «Цифровые технологии управления транспортными процессами»

Автор Заманов Евгений Альбертович

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ 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:	очно-заочная
Год начала подготовки	2020

<p style="text-align: center;">Одобрено на заседании Учебно-методической комиссии института Протокол № 4 30 апреля 2020 г. Председатель учебно-методической комиссии</p>  <p style="text-align: right;">Н.А. Клычева</p>	<p style="text-align: center;">Одобрено на заседании кафедры</p> <p>Протокол № 1 27 апреля 2020 г. Доцент</p>  <p style="text-align: right;">В.Е. Нутович</p>
---	--

Москва 2020 г.

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.

2.1. Наименования предшествующих дисциплин

Для изучения данной дисциплины необходимы следующие знания, умения и навыки, формируемые предшествующими дисциплинами:

2.1.1. Программирование :

Знания: Знать и понимать: существующие языки программирования и их методологии, принципы разработки процедурных программ и подпрограмм.

Умения: Уметь: формализовывать алгоритмическую и функциональную структуру программного обеспечения; разрабатывать блок схемы алгоритмов, подпрограмм.

Навыки: Владеть: основными инструментами разработки, отладки, тестирования и распространения программного обеспечения.

2.1.2. Технологии программирования:

Знания: принципы построения архитектуры программного обеспечения и виды архитектуры программного обеспечения, типовые решения, библиотеки программных модулей, шаблоны, классы объектов, используемые при разработке программного обеспечения, методы и средства проектирования

Умения: разрабатывать метрики (количественные показатели) работы ИС, анализировать исходные данные

Навыки: навыками разработки и согласования технических спецификаций на программные компоненты и их взаимодействие с архитектором программного обеспечения

2.1.3. Языки программирования высокого уровня:

Знания:

Умения:

Навыки:

2.2. Наименование последующих дисциплин

Результаты освоения дисциплины используются при изучении последующих учебных дисциплин:

2.2.1. Архитектура программного обеспечения

2.2.2. Разработка веб-приложений

3. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ (МОДУЛЮ), СООТНЕСЕННЫЕ С ПЛАНИРУЕМЫМИ РЕЗУЛЬТАТАМИ ОСВОЕНИЯ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

В результате освоения дисциплины студент должен:

№ п/п	Код и название компетенции	Ожидаемые результаты
1	ПКР-1 Способность выполнять работы и управлять работами по разработке архитектур и прототипов информационных систем (ИС)	<p>ПКР-1.1 Знать инструменты и методы проектирования архитектуры ИС; инструменты и методы верификации архитектуры ИС; возможности ИС; предметную область автоматизации; архитектуру, устройство и функционирование вычислительных систем; коммуникационное оборудование; сетевые протоколы; основы современных операционных систем; основы современных систем управления базами данных; устройство и функционирование современных ИС; современные стандарты информационного взаимодействия систем; программные средства и платформы инфраструктуры информационных технологий организаций; современные подходы и стандарты автоматизации организации (например, CRM, MRP, ERP..., ITIL, ITSM); системы классификации и кодирования информации, в том числе присвоение кодов документам и элементам справочников; отраслевую нормативную техническую документацию; источники информации, необходимой для профессиональной деятельности; современный отечественный и зарубежный опыт в профессиональной деятельности; основы бухгалтерского учета и отчетности организаций; основы налогового законодательства Российской Федерации; основы управленческого учета; основы международных стандартов финансовой отчетности (МСФО); основы управления торговлей, поставками и запасами; основы организации производства; основы управления персоналом, включая вопросы оплаты труда; основы финансового учета и бюджетирования; основы управления взаимоотношениями с клиентами и заказчиками (CRM); современные инструменты и методы управления организацией, в том числе методы планирования деятельности, распределения поручений, контроля исполнения, принятия решений; методологию ведения документооборота в организациях; инструменты и методы определения финансовых и производственных показателей деятельности организаций; культуру речи; правила деловой переписки.</p> <p>ПКР-1.2 Уметь проектировать архитектуру ИС; проверять (верифицировать) архитектуру ИС; кодировать на языках программирования; тестировать результаты прототипирования; проводить презентации; проводить переговоры.</p> <p>ПКР-1.3 Владеть навыками разработки архитектурной спецификации ИС; согласования архитектурной спецификации ИС с заинтересованными сторонами; разработки прототипа ИС в соответствии с требованиями; тестирования прототипа ИС на проверку корректности архитектурных решений; анализа</p>

№ п/п	Код и название компетенции	Ожидаемые результаты
		результатов тестов; принятие решения о пригодности архитектуры; согласования пользовательского интерфейса с заказчиком.
2	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	УК-2.1 Знать действующие правовые нормы. УК-2.2 Уметь выбирать оптимальные способы решения поставленных задач. УК-2.3 Владеть приемами решения поставленных задач исходя из действующих правовых норм, имеющихся ресурсов и ограничений.

4. ОБЪЕМ ДИСЦИПЛИНЫ (МОДУЛЯ) В ЗАЧЕТНЫХ ЕДИНИЦАХ И АКАДЕМИЧЕСКИХ ЧАСАХ

4.1. Общая трудоемкость дисциплины составляет:

5 зачетных единиц (180 ак. ч.).

4.2. Распределение объема учебной дисциплины на контактную работу с преподавателем и самостоятельную работу обучающихся

Вид учебной работы	Количество часов	
	Всего по учебному плану	Семестр 7
Контактная работа	48	48,15
Аудиторные занятия (всего):	48	48
В том числе:		
лекции (Л)	16	16
практические (ПЗ) и семинарские (С)	16	16
лабораторные работы (ЛР)(лабораторный практикум) (ЛП)	16	16
Самостоятельная работа (всего)	78	78
Экзамен (при наличии)	54	54
ОБЩАЯ трудоемкость дисциплины, часы:	180	180
ОБЩАЯ трудоемкость дисциплины, зач.ед.:	5.0	5.0
Текущий контроль успеваемости (количество и вид текущего контроля)	КП (1), ПК1, ПК2	КП (1), ПК1, ПК2
Виды промежуточной аттестации (экзамен, зачет)	ЭК	ЭК

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

№ п/п	Семестр	Тема (раздел) учебной дисциплины	Виды учебной деятельности в часах/ в том числе интерактивной форме						Формы текущего контроля успеваемости и промежуточной аттестации
			Л	ЛР	ПЗ	КСР	СР	Всего	
1	2	3	4	5	6	7	8	9	10
1	7	Раздел 1 Раздел 1. Процессы и потоки.	3	2	2		10	17	
2	7	Тема 1.1 Тема 1. Введение в проблематику параллельного программирования.	1					1	
3	7	Тема 1.2 Тема 2. Процесс. Состояния процесса Переключение контекста.	1				10	11	
4	7	Тема 1.3 Тема 3. Потоки, их создание и завершение. Переключение потоков.	1	2	2			5	
5	7	Раздел 2 Раздел 2. Процесс разработки параллельной программы.	3	6	2		14	25	
6	7	Тема 2.1 Тема 4. Основные проблемы при разработке параллельного алгоритма.	1	2	2		10	15	ПК1
7	7	Тема 2.2 Тема 5. Стратегии размещения задач.	1				2	3	
8	7	Тема 2.3 Тема 6. Модели решений при разработке параллельных алгоритмов.	1	4			2	7	
9	7	Раздел 3 Раздел 3. Межпроцессное взаимодействие	6	2	6		36	50	КП
10	7	Тема 3.1 Тема 7. Ресурсы и их характеристики.	1				10	11	
11	7	Тема 3.2 Тема 8. Проблемы взаимодействия процессов.	1		1		4	6	

№ п/п	Семестр	Тема (раздел) учебной дисциплины	Виды учебной деятельности в часах/ в том числе интерактивной форме						Формы текущего контроля успеваемости и промежуточной аттестации
			Л	ЛР	ПЗ	КСР	СР	Всего	
1	2	3	4	5	6	7	8	9	10
12	7	Тема 3.3 Тема 9. Межпроцессное взаимодействие.	1		1		5	7	
13	7	Тема 3.4 Тема 10. Примитивы синхронизации: критические секции. Алгоритм Петерсона.	1		1		2	4	
14	7	Тема 3.5 Тема 11. Примитивы синхронизации: Семафоры.	1		1		10	12	ПК2
15	7	Тема 3.7 Тема 12. Примитивы синхронизации: Мьютексы.	1	2	2		5	10	
16	7	Раздел 4 Раздел 4. Решение классических задач.	4	6	6		18	34	
17	7	Тема 4.1 Тема 13. Решение классической задачи межпроцессного взаимодействия («Обедающие философы», «Читатели и писатели», «Спящий парикмахер»)	2	2	2		10	16	
18	7	Тема 4.2 Тема 14. Вычисление значения определенного интеграла по методу адаптивной квадратуры в условиях разделяемой памяти	1	2	2		4	9	
19	7	Тема 4.3 Тема 15. Вычисление произведения матриц с использованием	1	2	2		4	9	

№ п/п	Семестр	Тема (раздел) учебной дисциплины	Виды учебной деятельности в часах/ в том числе интерактивной форме						Всего	Формы текущего контроля успеваемости и промежу- точной аттестации
			Л	ЛР	ПЗ	КСР	СР			
1	2	3	4	5	6	7	8	9	10	
		конвейерного алгоритма в условиях распределенной памяти								
20	7	Экзамен						54	ЭК	
21		Всего:	16	16	16		78	180		

4.4. Лабораторные работы / практические занятия

Лабораторные работы предусмотрены в объеме 16 ак. ч.

№ п/п	№ семестра	Тема (раздел) учебной дисциплины	Наименование занятий	Всего часов/ из них часов в интерактивной форме
1	2	3	4	5
1	7	Раздел 1. Процессы и потоки.	Тема 3. Потоки, их создание и завершение. Переключение потоков.	2
2	7	Раздел 2. Процесс разработки параллельной программы.	Тема 4. Основные проблемы при разработке параллельного алгоритма.	2
3	7	Раздел 2. Процесс разработки параллельной программы.	Тема 6. Модели решений при разработке параллельных алгоритмов.	4
4	7	Раздел 3. Межпроцессное взаимодействие	Тема 12. Примитивы синхронизации: Мьютексы.	2
5	7	Раздел 4. Решение классических задач.	Тема 13. Решение классической задачи межпроцессного взаимодействия («Обедающие философы», «Читатели и писатели», «Спящий парикмахер»)	2
6	7	Раздел 4. Решение классических задач.	Тема 14. Вычисление значения определенного интеграла по методу адаптивной квадратуры в условиях разделяемой памяти	2
7	7	Раздел 4. Решение классических задач.	Тема 15. Вычисление произведения матриц с использованием конвейерного алгоритма в условиях распределенной памяти	2
ВСЕГО:				16/ 0

Практические занятия предусмотрены в объеме 16 ак. ч.

№ п/п	№ семестра	Тема (раздел) учебной дисциплины	Наименование занятий	Всего часов/ из них часов в интерактивной форме
1	2	3	4	5
1	7	Раздел 1. Процессы и потоки.	Тема 3. Потоки, их создание и завершение. Переключение потоков.	2
2	7	Раздел 2. Процесс разработки параллельной программы.	Тема 4. Основные проблемы при разработке параллельного алгоритма.	2
3	7	Раздел 3. Межпроцессное взаимодействие	Тема 8. Проблемы взаимодействия процессов.	1

№ п/п	№ семестра	Тема (раздел) учебной дисциплины	Наименование занятий	Всего часов/ из них часов в интерактивной форме
1	2	3	4	5
4	7	Раздел 3. Межпроцессное взаимодействие	Тема 9. Межпроцессное взаимодействие.	1
5	7	Раздел 3. Межпроцессное взаимодействие	Тема 10. Примитивы синхронизации: критические секции. Алгоритм Петерсона.	1
6	7	Раздел 3. Межпроцессное взаимодействие	Тема 11. Примитивы синхронизации: Семафоры.	1
7	7	Раздел 3. Межпроцессное взаимодействие	Тема 12. Примитивы синхронизации: Мьютексы.	2
8	7	Раздел 4. Решение классических задач.	Тема 13. Решение классической задачи межпроцессного взаимодействия («Обедающие философы», «Читатели и писатели», «Спящий парикмахер»)	2
9	7	Раздел 4. Решение классических задач.	Тема 14. Вычисление значения определенного интеграла по методу адаптивной квадратуры в условиях разделяемой памяти	2
10	7	Раздел 4. Решение классических задач.	Тема 15. Вычисление произведения матриц с использованием конвейерного алгоритма в условиях распределенной памяти	2
ВСЕГО:				16/ 0

4.5. Примерная тематика курсовых проектов (работ)

Разработка приложения с параллельным решением задачи.

Разработка приложения с асинхронным доступом к данным.

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 завершённый объём 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. УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДЛЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)

№ п/п	№ семестра	Тема (раздел) учебной дисциплины	Вид самостоятельной работы студента. Перечень учебно-методического обеспечения для самостоятельной работы	Всего часов
1	2	3	4	5
1	7	Раздел 1. Процессы и потоки.	Тема 2. Процесс. Состояния процесса Переключение контекста.	10
2	7	Раздел 2. Процесс разработки параллельной программы.	Тема 4. Основные проблемы при разработке параллельного алгоритма.	10
3	7	Раздел 2. Процесс разработки параллельной программы.	Тема 5. Стратегии размещения задач.	2
4	7	Раздел 2. Процесс разработки параллельной программы.	Тема 6. Модели решений при разработке параллельных алгоритмов.	2
5	7	Раздел 3. Межпроцессное взаимодействие	Тема 7. Ресурсы и их характеристики.	10
6	7	Раздел 3. Межпроцессное взаимодействие	Тема 8. Проблемы взаимодействия процессов.	4
7	7	Раздел 3. Межпроцессное взаимодействие	Тема 9. Межпроцессное взаимодействие.	5
8	7	Раздел 3. Межпроцессное взаимодействие	Тема 10. Примитивы синхронизации: критические секции. Алгоритм Петерсона.	2
9	7	Раздел 3. Межпроцессное взаимодействие	Тема 11. Примитивы синхронизации: Семафоры.	10
10	7	Раздел 3. Межпроцессное взаимодействие	Тема 12. Примитивы синхронизации: Мьютексы.	5
11	7	Раздел 4. Решение классических задач.	Тема 13. Решение классической задачи межпроцессного взаимодействия («Обедающие философы», «Читатели и писатели», «Спящий парикмахер»)	10
12	7	Раздел 4. Решение классических задач.	Тема 14. Вычисление значения определенного интеграла по методу адаптивной квадратуры в условиях разделяемой памяти	4
13	7	Раздел 4. Решение классических задач.	Тема 15. Вычисление произведения матриц с использованием конвейерного алгоритма в условиях распределенной памяти	4
ВСЕГО:				78

7. ПЕРЕЧЕНЬ ОСНОВНОЙ И ДОПОЛНИТЕЛЬНОЙ ЛИТЕРАТУРЫ, НЕОБХОДИМОЙ ДЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ)

7.1. Основная литература

№ п/п	Наименование	Автор (ы)	Год и место издания Место доступа	Используется при изучении разделов, номера страниц
1	Программирование. Основы параллельного программирования. 3-е издание [Электронный ресурс]	Богачёв К.Ю.	М.: БИНОМ. Лаборатория знаний, 2015 http://e.lanbook.com/book/70745	Все разделы
2	Параллельное программирование. Модели и приемы	Федотов И.Е.	Москва : СОЛЮН-Пресс, 2017 https://e.lanbook.com/book/107666	Все разделы
3	Параллельное программирование для многоядерных процессоров	Сердюк Ю.П., Петров А.В.	Москва : ИНТУИТ, 2016 https://e.lanbook.com/book/100357	Все разделы

7.2. Дополнительная литература

№ п/п	Наименование	Автор (ы)	Год и место издания Место доступа	Используется при изучении разделов, номера страниц
4	Параллельное программирование с использованием OpenMP	Левин М.П.	Москва : ИНТУИТ, 2016 https://e.lanbook.com/book/100358	Все разделы
5	Параллельное программирование с использованием технологии MPI	Антонов А.С.	Москва : ИНТУИТ, 2016 https://e.lanbook.com/book/100359	Все разделы
6	Параллельное программирование с помощью языка C#	Туральчук К.А.	Москва : ИНТУИТ, 2016 https://e.lanbook.com/book/100360	Все разделы

8. ПЕРЕЧЕНЬ РЕСУРСОВ ИНФОРМАЦИОННО-ТЕЛЕКОММУНИКАЦИОННОЙ СЕТИ "ИНТЕРНЕТ", НЕОБХОДИМЫЕ ДЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ)

- <http://library.miit.ru/> - электронно-библиотечная система Научно-технической библиотеки МИИТ
- <https://ru.wikipedia.org> – Википедия
- <https://www.microsoft.com/ru-ru/learning/windows-training.aspx> - курсы Microsoft

9. ПЕРЕЧЕНЬ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ И ИНФОРМАЦИОННЫХ СПРАВОЧНЫХ СИСТЕМ, ИСПОЛЬЗУЕМЫХ ПРИ ОСУЩЕСТВЛЕНИИ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ (МОДУЛЮ)

Используемые информационные технологии:

? Прикладное программное обеспечение

Для выполнения лабораторных требуется следующее программное обеспечение:

? Microsoft Office

? Visual Studio

? Idea

10. ОПИСАНИЕ МАТЕРИАЛЬНО ТЕХНИЧЕСКОЙ БАЗЫ, НЕОБХОДИМОЙ ДЛЯ ОСУЩЕСТВЛЕНИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ (МОДУЛЮ)

Для лекционных занятий – наличие проектора и экрана. Для практических занятий – наличие персональных компьютеров вычислительного класса.

11. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ)

1. В лекционном курсе рассматриваются основные вопросы по данной дисциплине. Дополнительные вопросы, необходимые студентам при выполнении своих индивидуальных заданий, изучаются студентами самостоятельно и контролируются преподавателем.
2. Задания в рамках лабораторных работ выдаются студентам в начале семестра, чтобы студенты имели возможность самостоятельно изучить дополнительные теоретические сведения, необходимые им при выполнении индивидуальных заданий, и спланировать график выполнения заданий с учетом их специфики.
3. Прежде чем приступить к выполнению конкретного задания студент должен изучить: материалы лекций по теме задания; дополнительные материалы, относящиеся к специфике индивидуального задания; программные средства, используемые при выполнении задания.
4. Выполнение индивидуальных заданий и их сдача осуществляется по определенному графику и учитывается при периодической аттестации студентов.
5. Лекции по дисциплине, подготовленные в электронном виде, рекомендуется выдавать студентам в начале семестра с целью лучшего освоения материала и возможности досрочного изучения вопросов, необходимых для выполнения индивидуальных заданий.
6. Индивидуальные задания, требующие разработки сложных программных систем, могут выдаваться на группу студентов, но при этом необходимо контролировать знание каждым студентом всего задания в целом.
7. Для полноценного освоения дисциплины необходимо:
 - ? Посещение лекций и практических занятий;
 - ? Изучение лекционного материала;
 - ? Освоение теоретического материала, вынесенного на самостоятельное изучение, по предложенным источникам (литература, интернет-ресурсы);
 - ? Изучение программного обеспечения, необходимого, для выполнения индивидуальных заданий;
 - ? Консультации с преподавателем в ходе выполнения индивидуальных заданий и обсуждение промежуточных результатов выполнения индивидуальных заданий;
 - ? Своевременное выполнение индивидуальных заданий;
 - ? Своевременное предоставление отчетов по индивидуальным заданиям и защита выполненных работ.