

Документ подписан простой электронной подписью  
Информация о владельце:  
ФИО: Клочков Юрий Сергеевич  
Должность: и.о. ректора  
Дата подписания: 15.04.2024 15:21:07  
Уникальный программный ключ:  
4e7c4ea90328ec8e65c5d8058549a2538d7400d1

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ**

Федеральное государственное бюджетное  
образовательное учреждение высшего образования

**«ТОМЕНСКИЙ ИНДУСТРИАЛЬНЫЙ УНИВЕРСИТЕТ»**

**УТВЕРЖДАЮ**

Заместитель директора по УМР

\_\_\_\_\_ Т.А. Харитонова

«\_\_» \_\_\_\_\_ 2022 г.

**РАБОЧАЯ ПРОГРАММА**

дисциплины:

Параллельное программирование

направление подготовки:

01.03.02 Прикладная математика и информатика

направленность (профиль):

Прикладное программирование и компьютерные технологии

форма обучения:

очная

Рабочая программа разработана для обучающихся по направлению подготовки 01.03.02 Прикладная математика и информатика, направленность (профиль) Прикладное программирование и компьютерные технологии

Рабочая программа рассмотрена на заседании кафедры бизнес-информатики и математики

Заведующий кафедрой

\_\_\_\_\_ О.М. Барбаков  
(подпись)

Рабочую программу разработали:

Аханова М.А., доцент, к.с.н.

\_\_\_\_\_  
(подпись)

## **1. Цели и задачи освоения дисциплины**

Цель изучения дисциплины: формирование системы понятий и навыков в области параллельного программирования для параллельных систем различных классов (с общей памятью, с распределенной памятью, графические процессоры) и популярных технологий поддержки параллельных вычислений.

Задачи дисциплины:

- знакомство с архитектурой многопроцессорных систем, необходимых для понимания параллельных вычислений;
- изучение понятий параллелизма и общей техники распараллеливания программ;
- анализ проблем, сопровождающих разработку и отладку параллельных программ;
- изучение технологии параллельного программирования OpenMP для систем с общей памятью;
- изучение технологии CUDA как совокупности программно-аппаратных средств, обеспечивающих неграфические параллельные вычисления на графических процессорах;
- получение навыков использования технологии OpenCL параллельного программирования для гетерогенных вычислительных систем;
- знакомство с интерфейсом передачи сообщений MPI, являющегося общепризнанным стандартом для программирования параллельных вычислительных систем с распределенной памятью.
- развитие навыков самостоятельной научно-практической деятельности.

## **2. Место дисциплины в структуре ОПОП ВО**

Дисциплина относится к дисциплинам по выбору части учебного плана, формируемой участниками образовательных отношений.

Изучение дисциплины базируется на знаниях, полученных при изучении теоретической и прикладной информатики, программирования, алгоритмов и структур данных, объектно-ориентированного программирования, проектирования программного обеспечения, тестирования программного обеспечения, операционных систем и архитектур компьютерных сетей.

Необходимыми условиями для освоения дисциплины являются:

знание:

- основных принципов алгоритмизации и программирования;
- базовых алгоритмов и структур данных;
- архитектуры компьютерных систем;

умение:

- применять язык программирования в новых ситуациях;
- выявлять зависимости между итерациями циклических участков программы;

владение:

- навыками алгоритмизации и программирования;
- навыками тестирования и отладки компьютерных программ.

Основные положения дисциплины могут быть использованы при выполнении выпускной квалификационной работы, в профессиональной деятельности.

### 3. Результаты обучения по дисциплине

Процесс изучения дисциплины направлен на формирование следующих компетенций:

Таблица 3.1

Код и наименование компетенции	Код и наименование индикатора достижения компетенции (ИДК)	Код и наименование результата обучения по дисциплине
ПКС-1 Способен проектировать, разрабатывать, тестировать и документировать ПО	ПКС-1.2 Проектирует, разрабатывает и тестирует программное обеспечение с использованием современных средств и технологий на всех этапах жизненного цикла	Знать (31) компоненты программно-технических архитектур параллельных вычислительных систем;
		Знать (32) виды параллелизма и уровни распараллеливания;
		Знать (33) модель параллельной программы в технологии OpenMP;
		Знать (34) структуру памяти графического процессора и временные характеристики ее составных частей;
		Знать (35) модели платформы, памяти, исполнения и структуры параллельной программы гетерогенного компьютера;
		Знать (36) модель параллельной программы для вычислительной системы с распределенной памятью
		Уметь (У1) применять директивы OpenMP для создания параллельных регионов и распределения вычислительной работы между потоками;

		Уметь (У2) применять рекомендуемую технологией CUDA стратегию параллельного программирования графического процессора;
		Уметь (У3) применять рекомендуемую технологией OpenCL стратегию параллельного программирования гетерогенного компьютера;
		Уметь (У4) использовать основные группы функций интерфейса MPI для инициации библиотеки, получения сведений об ее окружении;
		Владеть (В1) способами преобразования циклов для ликвидации информационных зависимостей между итерациями;
		Владеть (В2) способами борьбы с гонками данных, блокировками и тупиковыми ситуациями;
		Владеть (В3) навыками разработки, компиляции и отладки параллельных программ с использованием технологии OpenMP;
		Владеть (В4) навыками разработки, компиляции и отладки параллельных программ с использованием технологии CUDA;
		Владеть (В5) навыками разработки, компиляции и отладки параллельных программ с использованием технологии OpenCL;
		Владеть (В6) навыками разработки, компиляции, профилирования и отладки параллельных программ с использованием технологии MPL

#### 4. Объем дисциплины

Общий объем дисциплины составляет 5 зачетных единиц, 180 часов.

Таблица 4.1

Форма обучения	Курс/семестр	Аудиторные занятия / контактная работа, час.			Самостоятельная работа, час.	Контроль, час	Форма промежуточной аттестации
		Лекции	Практические занятия	Лабораторные занятия			
Очная	4/8	40	-	26	87	27	Экзамен

#### 5. Структура и содержание дисциплины

##### 5.1. Структура дисциплины

**очная форма обучения (ОФО)**

Таблица 5.1

№ п/п	Структура дисциплины		Аудиторные занятия, час.			СРС, час.	Всего, час.	Код ИДК	Оценочные средства
	Номер раздела	Наименование раздела	Л.	Пр.	Лаб.				
1	1	Архитектура параллельных вычислительных систем. Введение в параллельное программирование	4	-	4	14	22	ПКС-1.2	Задания контрольной работы №1
2	2	OpenMP: параллельные вычисления на системах с общей памятью	8	-	6	18	32	ПКС-1.2	Вопросы к коллоквиуму №1, Задания контрольной работы №2
3	3	CUDA: параллельные вычисления на графическом процессоре	10	-	6	18	34	ПКС-1.2	Вопросы к коллоквиуму №2
4	4	OpenCL: параллельные вычисления на гетерогенном компьютере	8	-	6	18	32	ПКС-1.2	Вопросы к коллоквиуму №3
5	5	MPI: параллельные вычисления на системах с распределенной памятью	10	-	4	19	33	ПКС-1.2	Вопросы к коллоквиуму №4, Задания контрольной работы №3
6	1-5	Экзамен	-	-	-	27	27	ПКС-1.2	Вопросы к экзамену
Итого:			40	-	26	114	180	X	X

**Заочная форма обучения (ЗФО)**

Не реализуется

**Очно-заочная форма обучения (ОЗФО)**

Не реализуется

5.2. Содержание дисциплины.

5.2.1. Содержание разделов дисциплины (дидактические единицы)

**Раздел 1. Архитектура параллельных вычислительных систем. Введение в параллельное программирование.** Параллельные вычислительные системы. Основные

классы архитектур: ОКОД (Single Instruction Single Data, SISD), ОКМД (Single Instruction Multiple Data, SIMD), МКОД (Multiple Instruction Single Date, MISD), МКМД (Multiple Instruction Multiple Date, MIMD). Мультипроцессоры. Распараллеливание циклов. Распараллеливание гнезд циклов. Преобразования гнезд циклов. Проблемы разработки параллельных программ.

**Раздел 2. OpenMP: параллельные вычисления на системах с общей памятью.** Модель параллельной программы OpenMP. Директивы управления регионами и потоками. Директивы распределения вычислений. Директивы синхронизации. Переменные окружения. Библиотека функций OpenM.

**Раздел 3. CUDA: параллельные вычисления на графическом процессоре.** Архитектура CUDA. Аппаратные компоненты CUDA. Программные компоненты CUDA. CUDA-расширение языка C/C++. Технология разработки CUDA-программ. Библиотека CUDA-runtime, основные функции.

**Раздел 4. OpenCL: параллельные вычисления на гетерогенном компьютере.** Модель платформы. Модель памяти OpenCL. Модель исполнения OpenCL-программ. Модель структуры OpenCL-программ. Пример OpenCL-программ.

**Раздел 5. MPI: параллельные вычисления на системах с распределенной памятью.** Основные понятия интерфейса MPI. Функции инициализации и определения окружения. Обмен сообщениями типа «точка-точка». Коллективные операции взаимодействия процессов. Производные типы и упаковка/распаковка данных. Управление группами ветвей и коммутаторами. Управление виртуальными топологиями. Удаленный доступ к памяти (односторонние взаимодействия). Использование библиотеки MPE для анализа процессов взаимодействия ветвей программы.

## 5.2.2. Содержание дисциплины по видам учебных занятий

### Лекционные занятия

Таблица 5.2.1

№ п/п	Номер раздела дисциплины	Объем, час.			Тема лекции
		ОФО	ЗФО	ОЗФО	
1	1	4	-	-	Архитектура параллельных вычислительных систем. Введение в параллельное программирование
2	2	8	-	-	OpenMP: параллельные вычисления на системах с общей памятью
3	3	10	-	-	CUDA: параллельные вычисления на графическом процессоре

4	4	8	-	-	OpenCL: параллельные вычисления на гетерогенном компьютере
5	5	10	-	-	MPI: параллельные вычисления на системах с распределенной памятью
Итого:		40	-	-	X

### Практические занятия

Практические занятия не предусмотрены учебным планом

### Лабораторные работы

Таблица 5.2.2

№ п/п	Номер раздела дисциплины	Объем, час.			Тема лабораторного занятия
		ОФО	ЗФО	ОЗФО	
1	1	4	-	-	Архитектура параллельных вычислительных систем. Введение в параллельное программирование
2	2	6	-	-	OpenMP: параллельные вычисления на системах с общей памятью
3	3	6	-	-	CUDA: параллельные вычисления на графическом процессоре
4	4	6	-	-	OpenCL: параллельные вычисления на гетерогенном компьютере
5	5	4	-	-	MPI: параллельные вычисления на системах с распределенной памятью
Итого:		26	-	-	X

### Самостоятельная работа студента

Таблица 5.2.3

№ п/п	Номер раздела дисциплины	Объем, час.			Тема	Вид СРС
		ОФО	ЗФО	ОЗФО		
1	1	14	-	-	Архитектура параллельных вычислительных систем. Введение в параллельное программирование	Изучение теоретического материала, подготовка к контрольной работе
2	2	18	-	-	OpenMP: параллельные вычисления на системах с общей памятью	Изучение теоретического материала, подготовка к контрольной работе и к коллоквиуму
3	3	18	-	-	CUDA: параллельные вычисления на графическом процессоре	Изучение теоретического материала, подготовка к коллоквиуму
4	4	18	-	-	OpenCL: параллельные	Изучение теоретического



					вычисления на гетерогенном компьютере	материала, подготовка к коллоквиуму
5	5	19	-	-	МРІ: параллельные вычисления на системах с распределенной памятью	Изучение теоретического материала, подготовка к контрольной работе и к коллоквиуму
6	1 – 5	27	-	-	1-5	Изучение вопросов и подготовка к экзамену
Итого:		114	-	-	X	X

5.2.3. Преподавание дисциплины ведется с применением следующих видов образовательных технологий:

- визуализация учебного материала в PowerPoint и Google Colaboratory в диалоговом режиме (лекционные занятия);
- работа в малых группах (лабораторные занятия);
- индивидуальная работа (лабораторные занятия, срс).

## 6. Тематика курсовых работ/проектов

Не предусмотрены учебным планом.

## 7. Контрольные работы

- заочная форма обучения (ЗФО): не реализуется;
- очно-заочная форма обучения (ОЗФО): не реализуется.

## 8. Оценка результатов освоения дисциплины

8.1. Критерии оценивания степени полноты и качества освоения компетенций в соответствии с планируемыми результатами обучения приведены в Приложении 1.

8.2. Рейтинговая система оценивания степени полноты и качества освоения компетенций обучающихся очной формы обучения представлена в таблице 8.1.

Таблица 8.1

№ п/п	Виды мероприятий в рамках текущего контроля	Количество баллов
1	Контрольная работа №1	0 – 10
2	Коллоквиум №1	0 – 15
3	Контрольная работа №2	0 – 15
<b>ИТОГО за первую текущую аттестацию</b>		<b>0 – 40</b>

4	Коллоквиум №2	0 – 15
5	Коллоквиум №3	0 – 10
6	Коллоквиум №4	0 – 15
7	Контрольная работа №3	0 – 20
	ИТОГО за вторую текущую аттестацию	<b>0 – 60</b>
	<b>ВСЕГО</b>	<b>0 – 100</b>

## 9. Учебно-методическое и информационное обеспечение дисциплины

9.1. Перечень рекомендуемой литературы представлен в Приложении 2.

9.2. Современные профессиональные базы данных и информационные справочные системы:

- Электронный каталог/Электронная библиотека ТИУ <http://webirbis.tsogu.ru/>;
- Цифровой образовательный ресурс – библиотечная система IPR SMART — <https://www.iprbookshop.ru/>;
- Электронно-библиотечная система «Консультант студента» [www.studentlibrary.ru](http://www.studentlibrary.ru/);
- Электронно-библиотечная система «Лань» [https://e.lanbook.com](https://e.lanbook.com/);
- Образовательная платформа ЮРАЙТ [www.urait.ru](http://www.urait.ru/);
- Научная электронная библиотека ELIBRARY.RU [http://www.elibrary.ru](http://www.elibrary.ru/);
- Национальная электронная библиотека (НЭБ);
- ЭКБСОН – информационная система доступа к электронным каталогам библиотек сферы образования и науки;
- Библиотеки нефтяных вузов России:
- Электронная нефтегазовая библиотека РГУ нефти и газа им. Губкина <http://elib.gubkin.ru/>;
- Электронная библиотека Уфимского государственного нефтяного технического университета <http://bibl.rusoil.net/>;
- Библиотечно-информационный комплекс Ухтинского государственного технического университета УГТУ <http://lib.ugtu.net/books>;
- Электронная справочная система нормативно-технической документации «Технорматив».

9.3. Лицензионное и свободно распространяемое программное обеспечение, в т.ч. отечественного производства:

- Microsoft Windows;

- Microsoft Office Professional Plus;
- Dev-C ++ (свободно-распространяемое ПО);
- Visual Studio Code (свободно-распространяемое ПО);
- OpenCL (открытый язык для параллельных программ);
- OpenMP (открытый API);
- MPI (открытый языко-независимый протокол, используемый для программирования параллельных компьютеров);
- графический процессор CUDA (открытая платформа).

## 10. Материально-техническое обеспечение дисциплины

Помещения для проведения всех видов работы, предусмотренных учебным планом, укомплектованы необходимым оборудованием и техническими средствами обучения.

Таблица 10.1

### Обеспеченность материально-технических условий реализации ОПОП ВО

№ п/п	Наименование учебных предметов, курсов, дисциплин (модулей), практики, иных видов учебной деятельности, предусмотренных учебным планом образовательной программы	Наименование помещений для проведения всех видов учебной деятельности, предусмотренной учебным планом, в том числе помещения для самостоятельной работы, с указанием перечня основного оборудования, учебно-наглядных пособий	Адрес (местоположение) помещений для проведения всех видов учебной деятельности, предусмотренной учебным планом (в случае реализации образовательной программы в сетевой форме дополнительно указывается наименование организации, с которой заключен договор)
1	2	3	4
1.	Параллельное программирование	<p>Лекционные занятия: Учебная аудитория для проведения занятий лекционного типа; групповых и индивидуальных консультаций; текущего контроля и промежуточной аттестации. Оснащенность: Учебная мебель: столы, стулья. Моноблок - 1 шт., проектор - 1 шт., проекционный экран - 1 шт., акустическая система (колонки) - 4 шт., микрофон - 1 шт., документ-камера - 1 шт., телевизор - 2 шт.</p> <p>Лабораторные занятия: Учебная аудитория для проведения занятий семинарского типа (лабораторных занятий); групповых и индивидуальных консультаций; текущего контроля и промежуточной аттестации. Оснащенность:</p>	<p>625039, г. Тюмень, ул. Мельникайте, д. 70.</p> <p>625039, г. Тюмень, ул. Мельникайте, д. 70</p>

	Учебная мебель: столы, стулья. Моноблоки, проектор - 1 шт., проекционный экран - 1 шт., акустическая система (колонки) - 4 шт., микрофон - 1 шт., документ-камера - 1 шт., телевизор - 2 шт.	
--	--	--

## **11. Методические указания по организации СРС**

### 11.1. Методические указания по подготовке к лабораторным занятиям.

Важной формой самостоятельной работы студента является систематическая и планомерная подготовка к лабораторному занятию. После лекции студент должен познакомиться с планом лабораторных занятий и списком обязательной и дополнительной литературы, которую необходимо прочитать, изучить и законспектировать. Разъяснение по вопросам новой темы студенты получают у преподавателя в конце предыдущего лабораторного занятия.

Подготовка к лабораторному занятию требует, прежде всего, чтения рекомендуемых источников. Важным этапом в самостоятельной работе студента является повторение материала по конспекту лекции. Одна из главных составляющих внеаудиторной подготовки – работа с книгой. Она предполагает: внимательное прочтение, критическое осмысление содержания, обоснование собственной позиции по дискуссионным моментам, постановки интересующих вопросов, которые могут стать предметом обсуждения на практическом занятии.

В начале лабораторного занятия должен присутствовать организационный момент и вступительная часть. Преподаватель произносит краткую вступительную речь, где формулируются основные вопросы и проблемы, способы их решения в процессе работы.

В конце каждой темы подводятся итоги, предлагаются темы докладов, выносятся вопросы для самоподготовки.

Лабораторные занятия являются одной из важнейших форм обучения студентов: они позволяют студентам закрепить, углубить и конкретизировать знания по созданию и эксплуатации баз данных, подготовиться к научно-исследовательской деятельности. В процессе работы на лабораторных занятиях обучающийся должен совершенствовать умения и навыки самостоятельного анализа источников и научной литературы, что необходимо для научно-исследовательской работы.

Усвоенный материал необходимо научиться применять при решении поставленных задач.

Успешному осуществлению внеаудиторной самостоятельной работы способствует проведение коллоквиумов. Они обеспечивают непосредственную связь между студентом

и преподавателем (по ним преподаватель судит о трудностях, возникающих у студентов в ходе учебного процесса, о степени усвоения предмета, о помощи, какую надо указать, чтобы устранить пробелы в знаниях); они используются для осуществления контрольных функций.

#### 11.2. Методические указания по организации самостоятельной работы.

Самостоятельная работа является одной из важнейших форм изучения любой дисциплины. Она позволяет систематизировать и углубить теоретические знания, закрепить умения и навыки, способствует развитию умений пользоваться научной и учебно-методической литературой. Познавательная деятельность в процессе самостоятельной работы требует от студента высокого уровня активности и самоорганизованности.

В учебном процессе выделяют два вида самостоятельной работы: аудиторная и внеаудиторная.

Аудиторная самостоятельная работа по дисциплине выполняется на учебных занятиях под непосредственным руководством преподавателя и по его заданию.

Внеаудиторная самостоятельная работа студентов представляет собой логическое продолжение аудиторных занятий. Затраты времени на выполнение этой работы регламентируются рабочим учебным планом. Режим работы выбирает сам обучающийся в зависимости от своих способностей и конкретных условий.

Самостоятельная работа может осуществляться индивидуально или группами студентов в зависимости от цели, объема, конкретной тематики самостоятельной работы, уровня сложности, уровня умений студентов.

Самостоятельная работа включает в себя работу с конспектом лекций, изучение и конспектирование рекомендуемой литературы, изучение мультимедиа лекций, расположенных в свободном доступе, решение ситуационных (профессиональных) задач, проектирование и моделирование разных видов и компонентов профессиональной деятельности, научно-исследовательскую работу и др.

Контроль результатов внеаудиторной самостоятельной работы студентов может осуществляться в пределах времени, отведенного на обязательные учебные занятия по дисциплине и внеаудиторную самостоятельную работу студентов по дисциплине, может проходить в письменной, устной или смешанной форме.

Работа на лекции – это сложный процесс, который включает в себя такие элементы как слушание, осмысление и, собственно, конспектирование. Для того, чтобы лекция выполнила свое назначение, важно подготовиться к ней и ее записи еще до прихода преподавателя в аудиторию, поскольку в первые минуты лекции объявляется тема лекции,

формулируется ее основная цель. Без этого дальнейшее восприятие лекции становится сложным. Важно научиться слушать преподавателя во время лекции. Здесь не следует путать такие понятия как слышать и слушать. Слушание лекции состоит из нескольких этапов, начиная от слышания (первый шаг в процессе осмысленного слушания) и заканчивая оценкой сказанного.

Чтобы процесс слушания стал более эффективным, нужно разделять качество общения с лектором, научиться поддерживать непрерывное внимание к выступающему. Для оптимизации процесса слушания следует:

1. научиться выделять основные положения. Нельзя понять и запомнить все, что говорит выступающий, однако можно выделить основные моменты. Для этого необходимо обращать внимание на вводные слова, словосочетания, фразы, которые используются, как правило, для перехода к новым положениям, выводам и обобщениям;

2. во время лекции осуществлять поэтапный анализ и обобщение, услышанного. Необходимо постоянно анализировать и обобщать положения, раскрываемые в речи говорящего. Стараясь представить материал обобщенно, мы готовим надежную базу для экономной, свернутой его записи. Делать это лучше всего по этапам, ориентируясь на момент логического завершения одного вопроса (подвопроса, тезиса и т.д.) и перехода к другому;

3. готовность слушать выступление лектора до конца.

Слушание является лишь одним из элементов хорошего усвоения лекционного материала.

Поток информации, который сообщается во время лекции необходимо фиксировать, записывать – научиться вести конспект лекции, где формулировались бы наиболее важные моменты, основные положения, излагаемые лектором. Для ведения конспекта лекции следует использовать тетрадь. Ведение конспекта на листочках не рекомендуется, поскольку они не так удобны в использовании и часто теряются. При оформлении конспекта лекции необходимо оставлять поля, где студент может записать свои собственные мысли, возникающие параллельно с мыслями, высказанными лектором, а также вопросы, которые могут возникнуть в процессе слушания, чтобы получить на них ответы при самостоятельной проработке материала лекции, при изучении рекомендованной литературы или непосредственно у преподавателя в конце лекции.

Составляя конспект лекции, следует оставлять значительный интервал между строчками. Это связано с тем, что иногда возникает необходимость вписать в первоначальный текст лекции одну или несколько строчек, имеющих принципиальное значение и почерпнутых из других источников. Расстояние между строками необходимо

также для подчеркивания слов или целых групп слов (такое подчеркивание вызывается необходимостью привлечь внимание к данному месту в тексте при повторном чтении). Обычно подчеркивают определения, выводы.

Главным отличием конспекта лекции от текста является свертывание текста. При ведении конспекта удаляются отдельные слова или части текста, которые не выражают значимую информацию, а развернутые обороты речи заменяют более лаконичными или же синонимичными словосочетаниями. При конспектировании основную информацию следует записывать подробно, а дополнительные и вспомогательные сведения, примеры – очень кратко. Особенно важные моменты лекции, на которые следует обратить особое внимание лектор, как правило, читает в замедленном темпе, что позволяет сделать их запись дословной. Также важно полностью без всяких изменений вносить в тетрадь схемы, таблицы, чертежи и т.п., если они предполагаются в лекции. Для того, чтобы совместить механическую запись с почти дословным фиксированием наиболее важных положений, можно использовать системы условных сокращений. В первую очередь сокращаются длинные слова и те, что повторяются в речи лектора чаще всего. При этом само сокращение должно быть по возможности кратким.

**Планируемые результаты обучения для формирования компетенции и критерии их оценивания**

Дисциплина: **параллельное программирование**

Код, направление подготовки: **01.03.02 Прикладная математика и информатика**

Направленность (профиль): **Прикладное программирование и компьютерные технологии**

Код компетенции	Код, наименование ИДК	Код и наименование результата обучения по дисциплине	Критерии оценивания результатов обучения			
			1 – 2	3	4	5
ПКС-1	ПКС-1.2 Проектирует, разрабатывает и тестирует программное обеспечение с использованием современных средств и технологий на всех этапах жизненного цикла	Знать (31) компоненты программно-технических архитектур параллельных вычислительных систем	Не знает компоненты программно-технических архитектур параллельных вычислительных систем;	Демонстрирует знание отдельных компонент программно-технических архитектур параллельных вычислительных систем	Демонстрирует достаточные знания компонент программно-технических архитектур параллельных вычислительных систем	Демонстрирует исчерпывающие знания компонент программно-технических архитектур параллельных вычислительных систем
		Знать (32) виды параллелизма и уровни распараллеливания	Не знает виды параллелизма и уровни распараллеливания	Демонстрирует частичное знание видов параллелизма и уровней распараллеливания	Демонстрирует достаточные знания видов параллелизма и уровней распараллеливания	Демонстрирует исчерпывающие знания видов параллелизма и уровней распараллеливания
		Знать (33) модель параллельной программы в технологии OpenMP	Не знает модель параллельной программы в технологии OpenMP	Демонстрирует частичное знание модели параллельной программы в технологии OpenMP	Демонстрирует достаточные знания модели параллельной программы в технологии OpenMP	Демонстрирует исчерпывающие знания модели параллельной программы в технологии OpenMP
		Знать (34) структуру памяти графического процессора и временные характеристики ее составных частей	Не знает структуру памяти графического процессора и временные характеристики ее составных частей	Демонстрирует частичное знание структуры памяти графического процессора и временных характеристик ее составных частей	Демонстрирует достаточные знания структуры памяти графического процессора и временных характеристик ее составных частей	Демонстрирует исчерпывающие знания структуры памяти графического процессора и временных характеристик ее составных частей



		Знать (35) модели платформы, памяти, исполнения и структуры параллельной программы гетерогенного компьютера	Не знает модели платформы, памяти, исполнения и структуры параллельной программы гетерогенного компьютера;	Демонстрирует частичное знание платформы, памяти, исполнения и структуры параллельной программы гетерогенного компьютера	Демонстрирует достаточные знания модели платформы, памяти, исполнения и структуры параллельной программы гетерогенного компьютера	Демонстрирует исчерпывающие знания модели платформы, памяти, исполнения и структуры параллельной программы гетерогенного компьютера
		Знать (36) модель параллельной программы для вычислительной системы с распределенной памятью	Не знает модель параллельной программы для вычислительной системы с распределенной памятью	Демонстрирует частичное знание модели параллельной программы для вычислительной системы с распределенной памятью	Демонстрирует достаточные знания модели параллельной программы для вычислительной системы с распределенной памятью	Демонстрирует исчерпывающие знания модели параллельной программы для вычислительной системы с распределенной памятью
		Уметь (У1) применять директивы OpenMP для создания параллельных регионов и распределения вычислительной работы между потоками	Не умеет применять директивы OpenMP для создания параллельных регионов и распределения вычислительной работы между потоками	Частично умеет применять директивы OpenMP для создания параллельных регионов и распределения вычислительной работы между потоками	Умеет на хорошем уровне применять директивы OpenMP для создания параллельных регионов и распределения вычислительной работы между потоками	В совершенстве может применять директивы OpenMP для создания параллельных регионов и распределения вычислительной работы между потоками
		Уметь (У2) применять рекомендуемую технологией CUDA стратегию параллельного программирования графического процессора	Не умеет применять рекомендуемую технологией CUDA стратегию параллельного программирования графического процессора	Частично умеет применять рекомендуемую технологией CUDA стратегию параллельного программирования графического процессора	Умеет на хорошем уровне применять рекомендуемую технологией CUDA стратегию параллельного программирования графического процессора	В совершенстве может применять рекомендуемую технологией CUDA стратегию параллельного программирования графического процессора
		Уметь (У3)	Не умеет применять	Частично умеет	Умеет на хорошем	В совершенстве может

		применять рекомендуемую технологией OpenCL стратегию параллельного программирования гетерогенного компьютера	рекомендуемую технологией OpenCL стратегию параллельного программирования гетерогенного компьютера	применять рекомендуемую технологией OpenCL стратегию параллельного программирования гетерогенного компьютера	уровне применять рекомендуемую технологией OpenCL стратегию параллельного программирования гетерогенного компьютера	применять рекомендуемую технологией OpenCL стратегию параллельного программирования гетерогенного компьютера
	Уметь (У4)	использовать основные группы функций интерфейса MPI для инициации библиотеки, получения сведений об ее окружении	Не умеет использовать основные группы функций интерфейса MPI для инициации библиотеки, получения сведений об ее окружении	Частично умеет использовать основные группы функций интерфейса MPI для инициации библиотеки, получения сведений об ее окружении	Умеет на хорошем уровне использовать основные группы функций интерфейса MPI для инициации библиотеки, получения сведений об ее окружении	В совершенстве может использовать основные группы функций интерфейса MPI для инициации библиотеки, получения сведений об ее окружении
	Владеть (В1)	способами преобразования циклов для ликвидации информационных зависимостей между итерациями	Не владеет способами преобразования циклов для ликвидации информационных зависимостей между итерациями	Не достаточно владеет способами преобразования циклов для ликвидации информационных зависимостей между итерациями	На достаточном уровне владеет способами преобразования циклов для ликвидации информационных зависимостей между итерациями	В совершенстве владеет навыками способами преобразования циклов для ликвидации информационных зависимостей между итерациями
	Владеть (В2)	способами борьбы с гонками данных, блокировками и тупиковыми ситуациями	Не владеет способами борьбы с гонками данных, блокировками и тупиковыми ситуациями	Не достаточно владеет способами борьбы с гонками данных, блокировками и тупиковыми ситуациями	На достаточном уровне владеет способами борьбы с гонками данных, блокировками и тупиковыми ситуациями	В совершенстве владеет навыками способами борьбы с гонками данных, блокировками и тупиковыми ситуациями
	Владеть (В3)	навыками разработки, компиляции и отладки параллельных программ с использованием технологии OpenMP	Не владеет навыками разработки, компиляции и отладки параллельных программ с использованием технологии OpenMP	Не достаточно владеет навыками разработки, компиляции и отладки параллельных программ с использованием технологии OpenMP	На достаточном уровне владеет навыками разработки, компиляции и отладки параллельных программ с использованием технологии OpenMP	В совершенстве владеет навыками разработки, компиляции и отладки параллельных программ с использованием технологии OpenMP

		технологии OpenMP				
		Владеть (B4) навыками разработки, компиляции и отладки параллельных программ с использованием технологии CUDA	Не владеет навыками разработки, компиляции и отладки параллельных программ с использованием технологии CUDA	Не достаточно владеет навыками разработки, компиляции и отладки параллельных программ с использованием технологии CUDA	На достаточном уровне владеет навыками разработки, компиляции и отладки параллельных программ с использованием технологии CUDA	В совершенстве владеет навыками разработки, компиляции и отладки параллельных программ с использованием технологии CUDA
		Владеть (B5) навыками разработки, компиляции и отладки параллельных программ с использованием технологии OpenCL	Не владеет навыками разработки, компиляции и отладки параллельных программ с использованием технологии OpenCL	Не достаточно владеет навыками разработки, компиляции и отладки параллельных программ с использованием технологии OpenCL	На достаточном уровне владеет навыками разработки, компиляции и отладки параллельных программ с использованием технологии OpenCL	В совершенстве владеет навыками разработки, компиляции и отладки параллельных программ с использованием технологии OpenCL
		Владеть (B6) навыками разработки, компиляции, профилирования и отладки параллельных программ с использованием технологии MPL	Не владеет навыками разработки, компиляции, профилирования и отладки параллельных программ с использованием технологии MPL	Не достаточно владеет навыками разработки, компиляции, профилирования и отладки параллельных программ с использованием технологии MPL	На достаточном уровне владеет навыками разработки, компиляции, профилирования и отладки параллельных программ с использованием технологии MPL	В совершенстве владеет навыками разработки, компиляции, профилирования и отладки параллельных программ с использованием технологии MPL

## КАРТА

## обеспеченности дисциплины учебной и учебно-методической литературой

Дисциплина: **параллельное программирование**Код, направление подготовки: **01.03.02 Прикладная математика и информатика**Направленность (профиль): **Прикладное программирование и компьютерные технологии**

№ п/п	Название учебного, учебно-методического издания, автор, издательство, вид издания, год издания	Количество экземпляров в БИК	Контингент обучающихся, использующих указанную литературу	Обеспеченность обучающихся литературой, %	Наличие электронного варианта в ЭБС (+/-)
1	Бабичев, С. Л. Распределенные системы : учебное пособие для вузов / С. Л. Бабичев, К. А. Коньков. — Москва : Издательство Юрайт, 2022. — 507 с. — (Высшее образование). — ISBN 978-5-534-11380-8. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <a href="https://urait.ru/bcode/476142">https://urait.ru/bcode/476142</a>	ЭР*	30	100	+
2	Богачёв, К. Ю. Основы параллельного программирования : учебное пособие / К. Ю. Богачёв. — 4-е изд. — Москва : Лаборатория знаний, 2020. — 345 с. — ISBN 978-5-00101-758-5. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <a href="https://e.lanbook.com/book/135516">https://e.lanbook.com/book/135516</a>	ЭР*	30	100	+
3	Малявко, А. А. Параллельное программирование на основе технологий openmp, cuda, opencl, mpi : учебное пособие для вузов / А. А. Малявко. — 3-е изд., испр. и доп. — Москва : Издательство Юрайт, 2022. — 135 с. — (Высшее образование). — ISBN 978-5-534-14116-0. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <a href="https://urait.ru/bcode/492127">https://urait.ru/bcode/492127</a>	ЭР*	30	100	+

ЭР – электронный ресурс для автор. пользователей доступен через Электронный каталог/Электронную библиотеку ТИУ <http://webirbis.tsogu.ru/>