

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ФАКУЛЬТЕТ ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ

УТВЕРЖДАЮ
Декан факультета


« 08 » 2019 г. П. Р.


РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Б1.В.09 ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ И ПАРАЛЛЕЛЬНАЯ ОБРАБОТКА
ДАНЫХ

Направление подготовки — 01.03.02. Прикладная математика и информатика

Направленность(профиль) — Компьютерные технологии

Квалификация выпускника — бакалавр

Форма обучения — очная

Пенза, 2019

1. Цели освоения дисциплины

Целью освоения дисциплины «**Вычислительные системы и параллельная обработка данных**» является формирование фундаментальных знаний в области концептуальных решений организации параллелизма обработки информации на различных уровнях вычислительных систем, организации внутрипроцессорного параллелизма, многопроцессорных вычислительных систем.

Формируемые дисциплиной «**Вычислительные системы и параллельная обработка данных**» знания и умения готовят выпускника данной образовательной программы к выполнению следующих обобщенных трудовых функций (трудовых функций):

- Анализ требований к программному обеспечению (ПС 06.001 “Программисты”, ТФ D/01.6, D/03.6):
- Оценка архитектуры с точки зрения прослеживаемости требований:
 - согласованность с системными требованиями;
 - приспособленность стандартов и методов проектирования;
 - осуществимость функционирования и сопровождения;
 - осуществимость программных составных частей, полностью удовлетворяющих назначенным требованиям (ПС 06.003 “Архитектор программного обеспечения”, ТФ D/03.5)
- Разработка концепции системы (ПС 06.022 “Системный аналитик”, ТФ С/05.6);
- Организация оценки соответствия требованиям существующих систем и их аналогов (ПС 06.022 “Системный аналитик”, ТФ С/07.6).

2. Место дисциплины в структуре ОПОП бакалавриата

Дисциплина «**Вычислительные системы и параллельная обработка данных**» относится к части, формируемой участниками образовательных отношений, Блока 1 «Дисциплины (модули) ОПОП».

Для освоения дисциплины студенты используют знания, умения, навыки, сформированные в процессе изучения предметов «Основы информатики», «Алгоритмы и алгоритмические языки», «Языки и методы программирования», «Линейная алгебра и аналитическая геометрия», «Технологии параллельного программирования», «Численные методы».

Освоение данной дисциплины является основой для последующего прохождения практик: Производственной (проектной (проектно-технологической)) практики, Производственной (преддипломной) практики, выполнению и защите выпускной квалификационной работы.

3. Результаты освоения дисциплины «Компьютерная графика»

Процесс изучения дисциплины направлен на формирование элементов следующих компетенций в соответствии с ФГОС ВО по данному направлению:

Коды компетенции	Наименование компетенции	Индикатор достижения компетенции	В результате освоения дисциплины обучающийся должен:
УК-2	Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм,	<i>УК– 2.2. Предлагает способы решения поставленных задач и ожидаемые результаты; оценивает предложенные способы с точки зрения соответствия цели проекта</i>	Знать основные факты, концепции, принципы теорий, связанных с разработкой параллельных и распределённых программ обработки данных Уметь: использовать базовые знания естественных наук, математики и информатики в

	<i>имеющихся ресурсов и ограничений</i>		процессе разработки параллельных и распределённых программ обработки данных Владеть: навыками разработки параллельных и распределённых программ обработки данных с использованием базовых знаний естественных наук, математики и информатики
<i>УК-6</i>	Способен управлять своим временем, выстраивать и реализовывать траекторию саморазвития на основе принципов образования в течение всей жизни	<i>УК – 6.1. Использует инструменты и методы управления временем при выполнении конкретных задач, проектов, при достижении поставленных целей</i>	Знать: современные способы и средства приобретения с помощью информационных технологий новых знаний и умений в области параллельной обработки данных и использования их в практической деятельности Уметь: приобретать с помощью информационных технологий и использовать в практической деятельности новые знания и умения области параллельной обработки данных Владеть: профессиональными навыками работы с информационными и компьютерными технологиями в научной и познавательной деятельности области параллельной обработки данных
<i>ПК-1</i>	Способен к анализу требований и разработке вариантов реализации систем обработки данных	<i>ПК-1.2. Разрабатывает варианты реализации систем обработки данных</i>	Знать: основы параллельной обработки данных при разработке алгоритмических и программных решений в области системного и прикладного программирования Уметь: решать задачи разработки на профессиональном уровне алгоритмических и программных решений в области разработки параллельных и распределённых программ обработки данных Владеть: практическими навыками разработки алгоритмических и программных решений в

			области параллельных и распределённых программ обработки данных
<i>ПК-3.</i>	ПК-3. Способен применять в профессиональной деятельности современные языки программирования и методы параллельной обработки данных, операционные системы, электронные библиотеки и пакеты программ, сетевые технологии	<i>ПК-3.2. Формулирует решение прикладных задач в профессиональной сфере деятельности</i>	Знать: базовые алгоритмы функционирования компьютерных систем параллельной обработки данных Уметь: формализовать поставленные задачи и разрабатывать алгоритмы для компьютерных систем параллельной обработки данных Владеть практическими навыками формализации и алгоритмизации поставленных задач в области параллельной обработки данных

4. Структура и содержание дисциплины «Вычислительные системы и параллельная обработка данных»

4.1. Структура дисциплины

Общая трудоемкость дисциплины составляет 4 зачетные единицы, 144 часа.

№ п/п	Наименование разделов и тем дисциплины	семестр	Недели семестра	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)									Формы текущего контроля успеваемости (по неделям семестра)			
				Контактная работа				Самостоятельная работа					Опрос на лабораторном занятии	Проверка выполнения индивидуального задания	Проверка отчёта выполнения лабораторной работы	Курсовой проект
				Всего	Лекция	Лабораторные занятия	Другие виды контактной работы	Всего	Курсовой проект	Подготовка к аудиторным занятиям	Выполнение индивидуального задания	Подготовка отчёта лабораторной работы				
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
1.	Раздел 1. Модели параллельных программ	6	1	2,3	2		0,3	2,55		2,55						
1.1.	Тема 1.1. Модели параллельных программ		1	2,3	2		0,3	2,55		2,55						
2.	Раздел 2. Базовые параллельные методы обработки данных	6	1-7	18,8	4	14	0,8	25	10	2	10	3				
2.1.	Тема 2.1. Базовые параллельные методы обработки данных		3-5	4,3	4		0,3	2		2						
2.2.	Лабораторная работа 1. Распараллеливание операций над матрицами		1-2	4		4		4			3	1	1	2	3	
2.3.	Лабораторная работа 2. Реализация распределённой обработки связанных списков		3-5	6		6		5			4	1	3	5	6	

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
2.4	Лабораторная работа 3. Реализация распределённого вычисления интеграла по правилу трапеции		6-7	4		4		4			3	1	6	7	8	
2.5	Изучение литературных источников по выбранной теме проекта		2-4	0,5			0,5	10	10							5
3	Раздел 3. Раздел 3. Сортировка данных	6	6-12	7,3	2	4	1,3	27	22	2	3	1				
3.1	Тема 3.1. Сортировка данных		7	2,3	2		0,3	2		2						
3.2	Лабораторная работа 4. Реализация распределённой сортировки элементов двумерного массива		8-9	4		4		4			3	1	8	9	10	
3.3	Разработка алгоритма		5-7	0,5			0,5	8	8							7
3.4	Разработка программы		8-12	0,5			0,5	14	14				6	7	8	13
4	Раздел 4. Раздел 4. Генерация псевдослучайных чисел	6	9-13	2,25	2		0,25	2		2						
4.1	Тема 4.1. Генерация псевдослучайных чисел		9	2,25	2		0,25	2		2						
5	Раздел 5. Декомпозиция сеточных графов		10-11	6,3	2	4	0,3	7		2	3	1				5
5.1	Тема 5.1 Декомпозиция сеточных графов		11	2,3	2		0,3	2		2						
5.2	Лабораторная работа 5. Реализация параллельных алгоритмов, использующих графы		10-11	4		4		4			3	1	10	11	12	
6	Раздел 6. Динамическая балансировка загрузки процессоров		12-14	8,8	2	6	0,8	13	6	2	4	1	10	11	12	
6.1	Тема 6.1. Динамическая балансировка загрузки процессоров		13	2,3	2		0,3	2		2			12	13	14	

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	
6.2	Лабораторная работа 6. Реализация параллельных алгоритмов обработки изображений		12-14	6		6		5			4	1	12	14	15		
6.3	Отладка программы		13-14	0,5			0,5	6	6							14	
7	Раздел 7 Анализ производительности параллельных приложений		15-17	9,7	3	6	0,7	12	5	2	4	1	16	16	17		
7.1	Тема 7.1. Анализ производительности параллельных приложений		15-17	3,2	3		0,2	2		2							
7.2	Лабораторная работа 7. Реализация распределённого решения системы линейных алгебраических уравнений методом Якоби		15-17			6		5			4	1	15	16	17		
7.3	Оформление пояснительной записки		15	0,5			0,5	5	5							16	
	<i>Курсовой проект</i>			2,5			2,5		45								
	Общая трудоемкость, в часах			55,45	17	34	4,45	88,55	45	6,85	18	7	Промежуточная аттестация				
												Форма					Семестр
												Зачет					
												Зачет с оценкой					6
														Экзамен			

4.2. Содержание дисциплины

Раздел 1. Модели параллельных программ

Тема 1.1. Модели параллельных программ

Вычислительные системы с распределённой памятью. Вычислительные системы с общей памятью. Гибридные архитектуры. Модель выполнения параллельной программы на распределённой памяти. Модель выполнения параллельной программы на общей памяти. Средства взаимодействия последовательных процессов.

Раздел 2. Базовые параллельные методы обработки данных

Тема 2.1. Базовые параллельные методы обработки данных

Метод сдваивания. Быстрый алгоритм выбора частных сумм. Барьерная синхронизация на основе синхронных обменов. Определение потока. Метод геометрического параллелизма. Метод конвейерного параллелизма. Метод коллективного решения.

Лабораторная работа 1. Распараллеливание операций над матрицами

Лабораторная работа 2. Реализация распределённой обработки связанных списков

Лабораторная работа 3. Реализация распределённого вычисления интеграла по правилу трапеции

Раздел 3. Сортировка данных

Тема 3.1. Сортировка данных

Последовательные алгоритмы сортировки. Свойства последовательных алгоритмов сортировки. Масштабируемые алгоритмы сортировки. Сети сортировки. Сеть чётно-нечётной сортировки. Сеть обменной сортировки со слиянием Бэтчера. Сортировка больших массивов. Сравнение алгоритмов сортировки.

Лабораторная работа 4. Реализация распределённой сортировки элементов двумерного массива

Раздел 4. Генерация псевдослучайных чисел

Тема 4.1. Генерация псевдослучайных чисел

Требования к генераторам псевдослучайных чисел. Линейно-конгруэнтные генераторы. М-последовательности. Проверка примитивности полиномов. Тестирование генераторов псевдослучайных чисел.

Раздел 5. Декомпозиция сеточных графов

Тема 5.1. Декомпозиция сеточных графов

Критерии декомпозиции графов: классический критерий декомпозиции графов, выделение обоснованных доменов, минимизация максимальной степени доменов, обеспечение связности графов каждого из доменов. Декомпозиция на основе исходной нумерации узлов. Рекурсивная бисекция. Декомпозиция регулярных графов. Методы декомпозиции произвольных графов: иерархическая декомпозиция, спектральная бисекция, алгоритм инкрементного роста. Декомпозиция больших сеток.

Лабораторная работа 5. Реализация параллельных алгоритмов, использующих графы

Раздел 6. Динамическая балансировка загрузки процессоров

Тема 6.1. Динамическая балансировка загрузки процессоров

Стратегии балансировки загрузки. Метод диффузионной балансировки. Адаптивное интегрирование: последовательные алгоритмы, параллельные алгоритмы.

Лабораторная работа 6. Реализация параллельных алгоритмов обработки изображений

Раздел 7. Анализ производительности параллельных приложений

Тема 7.1. Анализ производительности параллельных приложений

Мотивы и основные концепции производительности: параллелизм против производительности, потоки и процессы, латентность и скорость обработки. Причины потери производительности: накладные расходы, нераспараллеливаемый код, состязания, время простоя. Компромиссы производительности: взаимодействие в зависимости от вычислений, параллелизм в зависимости от памяти, накладные расходы в зависимости от параллелизма. Измерение производительности. Масштабируемая производительность.

Лабораторная работа 7. Реализация распределённого решения системы линейных алгебраических уравнений методом Якоби

5. Образовательные технологии

В ходе освоения дисциплины «Вычислительные системы и параллельная обработка данных» при проведении аудиторных занятий используется образовательная технология, предусматривающая такие методы и формы изучения материала как лекция, лабораторное занятие, включающие в том числе активные и интерактивные формы занятий:

- проведение интерактивной лекции (Тема 2.1. «Базовые параллельные методы обработки данных»);
- проведение лекции проблемного характера (Тема 7.1. «Анализ производительности параллельных приложений»);
- проведение работы в парах (Лабораторная работа 7. «Реализация распределённого решения системы линейных алгебраических уравнений методом Якоби»).

Занятия, проводимые в интерактивной форме составляют 25 % от общего количества аудиторных занятий.

Самостоятельная работа студентов подразумевает работу под руководством преподавателя (консультации, помощь в написании и отладки программ и др.) и индивидуальную работу студента, выполняемую в том числе в компьютерном классе с выходом в Интернет на факультете вычислительной техники университета.

При реализации образовательных технологий используются следующие виды самостоятельной работы:

- работа с конспектом лекции (обработка текста);
- подготовка к лабораторной работе;
- выполнение индивидуального задания по лабораторным работам;
- выполнение курсового проекта;
- поиск информации в сети «Интернет» и литературе;
- подготовка к сдаче зачёта.

В целях реализации индивидуального подхода к обучению студентов, осуществляющих учебный процесс по собственной траектории в рамках индивидуального рабочего плана, изучение данной дисциплины базируется на следующих возможностях: обеспечение внеаудиторной работы со студентами в том числе в электронной образовательной среде с использованием соответствующего программного оборудования, дистанционных форм обучения. возможностей интернет-ресурсов, индивидуальных консультаций и т.д.

В качестве других видов контактной работы запланированы консультации при подготовке и проведении текущей и промежуточной аттестации.

	2.3	<i>Подготовка к лабораторной работе. Выполнение индивидуального задания Оформление отчета по лабораторной работе</i>	Изучить подходы распараллеливания операций со связными списками Разработать программу согласно индивидуальному заданию <i>Пример индивидуального задания</i> Вычислить сумму элементов связного списка, не являющиеся простыми Подготовить отчёт по лабораторной работе	5, 6, 10, 12	4 1
	2.5	<i>Выполнение курсового проекта</i>	Изучить принципы построения параллельных алгоритмов, разработки по ним параллельных приложений, литературные источники согласно задания на курсовой проект	3, 4, 7, 8, 11	10
5-7	2.4	<i>Подготовка к лабораторной работе. Выполнение индивидуального задания Оформление отчета по лабораторной работе</i>	Изучить подходы распараллеливания операций вычисления определённых интегралов Разработать программу согласно индивидуальному заданию <i>Пример индивидуального задания</i> Разработать программу приближённого вычисления определённого интеграла с точностью ϵ с помощью стратегии «управляющий – рабочие», используя алгоритм адаптивной квадратуры. Для оценки точности использовать правило Рунге. Подготовить отчёт по лабораторной работе	4,6,8,11	3 1
	3.3	<i>Выполнение курсового проектирования</i>	Разработка параллельного алгоритма обработки данных согласно задания на курсовой проект	4,5,6,7,8,10,12	8

	3.1	<i>Подготовка к аудиторному занятию</i>	Рассмотреть последовательные алгоритмы сортировки, масштабируемые алгоритмы последовательных алгоритмов, сети сортировок, сортировку больших массивов	4,7,8,11	2
8-10	3.2	<i>Подготовка к лабораторной работе. Выполнение индивидуального задания Оформление отчета по лабораторной работе</i>	Изучить принципы параллельной сортировки элементов массива Разработать программу согласно индивидуальному заданию <i>Пример индивидуального задания</i> Упорядочить по неубыванию каждую строку матрицы $A(m, n)$, а после этого перестановкой строк упорядочить всю матрицу по неубыванию элементов первого столбца. Подготовить отчёт по лабораторной работе	1, 5, 6, 10	3 1
	4.1	<i>Подготовка к аудиторному занятию</i>	Рассмотреть требования к генераторам случайных чисел, линейно-конгруэнтные генераторы, способы тестирования генераторов псевдослучайных чисел.	4,7,8,11	2
11-12	5.2	<i>Подготовка к лабораторной работе. Выполнение индивидуального задания Оформление отчета по лабораторной работе</i>	Изучить принципы распараллеливания операций над графами Разработать программу согласно индивидуальному заданию <i>Пример индивидуального задания</i> Разработать программу, проверяющую заданный оргграф на наличие циклов и при их наличии выводящую каждый цикл	5, 6, 10,12	3

			в виде последовательности вершин циклического пути Подготовить отчёт по лабораторной работе		1	
	3.4	<i>Выполнение курсового проектирования</i>	Разработка параллельной программы обработки данных согласно задания на курсовой проект	4,5,6,7,8,10,12	14	
	5.1	<i>Подготовка к аудиторному занятию</i>	Рассмотреть критерии декомпозиции графов, декомпозицию регулярных графов, декомпозицию больших сеток, методы декомпозиции произвольных графов.	4,7,8,11	2	
13-15	6.1	<i>Подготовка к аудиторному занятию</i>	Рассмотреть стратегии балансировки загрузки, метод диффузионной балансировки, последовательные и параллельные алгоритмы адаптивного интегрирования.	4,7,8,11	2	
	6.2	<i>Подготовка к лабораторной работе. Выполнение индивидуального задания Оформление отчета по лабораторной работе</i>	Изучить принципы распараллеливания алгоритмов обработки изображений Разработать программу согласно индивидуальному заданию <i>Пример индивидуального задания</i> Разработать программу, выполняющую сглаживание цветного изображения. Подготовить отчёт по лабораторной работе	5, 6, 10,12	4 1	
	6.3	<i>Выполнение курсового проектирования</i>	Отладить разработанную программу. Протестировать её работу при различных входных данных	4,5,6,7,8,10,12	6	
15-17	7.1	<i>Подготовка к аудиторному занятию</i>	Рассмотреть основные концепции производительности, причины потери производительности, компромиссы	4, 7, 8, 11	2	

			производительности, измерение производительности.		
7.2	<i>Подготовка к лабораторной работе. Выполнение индивидуального задания Оформление отчета по лабораторной работе</i>	Изучить функции парного взаимодействия интерфейса MPI Разработать программу согласно индивидуальному заданию <i>Пример индивидуального задания</i> Разработать программу решения системы линейных алгебраических уравнений $A^2X=Y^T$ Подготовить отчёт по лабораторной работе	1, 5, 6, 10, 12	4	1
7.3	<i>Выполнение курсового проектирования</i>	Оформить пояснительную записку курсового проекта	4,5,6,7,8,10,12	5	

6.2. Методические указания по организации самостоятельной работы студентов

Планируются следующие виды самостоятельной работы:

- подготовка к лабораторным работам занятиям,
- выполнение индивидуального задания,
- оформление отчётов по лабораторным работам,
- выполнение курсового проекта,
- оформление пояснительной записки к курсовому проекту,
- подготовка к экзамену,
- работа с конспектом лекций и изучение литературы при подготовке к экзаменам.

Подготовка к лабораторным занятиям проводится посредством изучения курса лекций, дополнительной литературы, Интернет-ресурсов.

Задание к выполнению каждой лабораторной работы состоит с общей части, которая сформулирована в разделе "Задание к выполнению" и уточнения варианта, который приведён в разделе "Варианты заданий". Студент должен заранее ознакомиться со своим заданием и, если у него возникают какие-либо вопросы относительно задания, поставить эти вопросы преподавателю до начала работы.

Выполнение начинается с разработки алгоритма решения задачи. На этом этапе должны быть детально проанализированы условия задания и разработанный алгоритм программы. Хотя представление схемы алгоритма не является обязательным элементом проектирования программы, но схема является удобным инструментом для осмысления задачи и оптимизации решения. Графическое представление алгоритма не освобождает от необходимости его текстового описания, в котором должны быть обоснованы ключевые алгоритмические решения. При разработке алгоритма следует уделять внимание его упрощению, минимизации объёма вычислений, удалению лишних операций и т.п.

Написание текста программы начинается с определения переменных, которые необходимы для функционирования алгоритма. Большая часть переменных может быть определена ещё на этапе проектирования схемы алгоритма.

Если схема алгоритма сделана достаточно тщательно, написание кодовой части программы сводится к записи каждого элемента схемы командой языка ассемблера. Подготовленный текст программы следует набрать в текстовом редакторе и сохранить его в файле, который затем необходимо скомпилировать, чтобы получить исполняемый файл.

Отчёт к лабораторной работе должен содержать:

- Тему работы
- Цель работы
- Задание для выполнения, включая индивидуальное задание
- Описание алгоритма программы, (при необходимости – со схемой алгоритма)
- Описание переменных и структур данных, которые применяются в программе
- Описание ключевых программных решений, принятых при реализации алгоритма в тексте программы
- Текст программы
- Результат работы программы
- Выводы

Курсовой проект должен соответствовать следующим требованиям:

- соответствовать компетенциям, формируемым у обучающихся в рамках соответствующей дисциплины или модуля;
- тема проекта, его цели и задачи должны быть связаны с решением проблем исследования;
- быть выполненными на достаточном теоретическом уровне и соответствовать установленному в университете пределу допустимого уровня заимствований для работ подобного типа;
- включать анализ не только теоретического, но и эмпирического материала;
- основываться на результатах самостоятельного исследования или расчетов;
- иметь самостоятельные выводы, изложенные в заключительной части пояснительной записки.

Пояснительная записка к курсовому проекту имеет следующую структуру:

- титульный лист;
- задание на курсовое проектирование;
- оглавление;
- введение;
- основной текст;
- заключение;
- список используемых источников;
- приложения.

Подготовка к экзамену проводится посредством изучения курса лекций, изучения дополнительной литературы, Интернет-ресурсов.

6.3. Материалы для проведения текущего и промежуточного контроля знаний студентов

Контроль освоения компетенций

№ п\п	Вид контроля	Контролируемые темы (разделы)	Компетенции, компоненты которых контролируются
1	Текущий: проверка выполнения индивидуального задания на лабораторной работе	Разделы 1 – 7	УК-2,6; ПК-1, ПК-3

2	Текущий: собеседование при защите лабораторных работ	Разделы 1 – 7	УК-2,6; ПК-1, ПК-3
3	Текущий: проверка выполнения курсового проекта	Разделы 1 – 7	УК-2,6; ПК-1, ПК-3
4	Промежуточный: собеседование при защите курсового проекта	Разделы 1 – 7	УК-2,6; ПК-1, ПК-3
5	Промежуточный: Зачет по результатам выполнения лабораторных работ	Разделы 1 – 4	УК-2,6; ПК-1, ПК-3

Материалы для проведения текущего контроля знаний и промежуточной аттестации составляют отдельный документ – Фонд оценочных средств по дисциплине «Разработка параллельного и распределённого программного обеспечения».

Демонстрационные варианты оценочных средств для каждого вида контроля смотри в ЭИОС (<http://moodle.pnzgu.ru>) в разделе Оценочные средства по дисциплине в курсе «Разработка параллельного и распределённого программного обеспечения».

Перечень примерных тем курсовых проектов

Примерные темы курсовых проектов

1. Напишите многопоточную рекурсивную программу для реализации алгоритма быстрой сортировки массива из n чисел. Сравните время выполнения программы с аналогичной последовательной программой.
2. Напишите многопоточную программу для реализации алгоритма умножения двух матриц размерностью $n \times n$. Сравните время выполнения программы с аналогичной последовательной программой.
3. Напишите многопоточную программу для реализации алгоритма транспонирования матрицы размерностью $n \times n$, при этом количество потоков кратно n . Сравните время выполнения программы с аналогичной последовательной программой.
4. Даны целочисленные массивы $a[m]$ и $b[n]$, причем значения в каждом массиве не повторяются. Разработайте многопоточную программу подсчета числа разных значений в *обоих* массивах. Сравните время выполнения программы с аналогичной последовательной программой.
5. Дано бинарное дерево, представленное связанной структурой: целочисленного значения и указателей на левое и правое поддеревья. Напишите рекурсивную многопоточную программу для вычисления суммы значений всех узлов дерева. Сравните время выполнения программы с аналогичной последовательной программой.

Перечень примерных вопросов к зачёту

1. Вычислительные системы с распределённой памятью.
2. Вычислительные системы с общей памятью.

3. Гибридные вычислительные архитектуры.
4. Модель выполнения параллельной программы на распределённой памяти.
5. Модель выполнения параллельной программы на общей памяти.
6. Средства взаимодействия последовательных процессов.
7. Барьерная синхронизация на основе синхронных обменов.
8. Метод геометрического параллелизма.
9. Метод конвейерного параллелизма.
10. Метод коллективного решения.
11. Свойства последовательных алгоритмов сортировки.
12. Масштабируемые алгоритмы сортировки.
13. Сеть чётно-нечётной сортировки.
14. Сеть обменной сортировки со слиянием Бэтчера.
15. Сортировка больших массивов.
16. Линейно-конгруэнтные генераторы псевдослучайных чисел.
17. Тестирование генераторов псевдослучайных чисел.
18. Критерии декомпозиции графов.
19. Декомпозиция регулярных графов.
20. Методы декомпозиции произвольных графов.
21. Декомпозиция больших сеток.
22. Стратегии балансировки загрузки процессоров.
23. Метод диффузионной балансировки процессоров.
24. Мотивы и основные концепции производительности приложений.
25. Причины потери производительности приложений.
26. Компромиссы производительности.
27. Измерение производительности.
28. Масштабируемая производительность.

7. Учебно-методическое и материально-техническое обеспечение дисциплины «Компьютерная графика»

а) учебная литература:

1. Ремонтов А.П., Ремонтова Л.В. Компьютерная графика: учебное пособие – Пенза, изд-во Саад Ю. Итерационные методы для разреженных линейных систем. В 2-х т.: учеб. пособие. Т. 1. — М.: Изд-во Моск. ун-та, 2013. — 344 с. (30 экземпляров) http://kleopatra.pnzgu.ru/cgi-bin/irbis64r_91/cgiirbis_64.exe?P21DBN=KATL&I21DBN=KATL_PRINT&S21FMT=fullw_print&C21COM=F&Z21MFN=15974
2. Кольчугина Е.А. Параллельное программирование: Учебное пособие. – Пенза: Изд-во Пенз. гос. ун-та, 2005. (10 экземпляров) http://kleopatra.pnzgu.ru/cgi-bin/irbis64r_91/cgiirbis_64.exe?P21DBN=KATL&I21DBN=KATL_PRINT&S21FMT=fullw_print&C21COM=F&Z21MFN=4590
3. Линев А. В., Боголепов Д. К., Бастраков С. И. Технология параллельного программирования для процессоров новых архитектур: учебник. – М.: Изд-во Моск. ун-та, 2010. (31 экземпляр) http://kleopatra.pnzgu.ru/cgi-bin/irbis64r_91/cgiirbis_64.exe?P21DBN=KATL&I21DBN=KATL_PRINT&S21FMT=fullw_print&C21COM=F&Z21MFN=13978

4. Кепнер Дж. Параллельное программирование в среде MATLAB для многоядерных и многоузловых вычислительных машин: учеб. пособие / Дж. Кепнер ; под ред. Д. В. Дуброва ; авт. предисл. В. А. Садовничий. - М. : Изд-во Моск. ун-та, 2013. (30 экземпляров) http://kleopatra.pnzgu.ru/cgi-bin/irbis64r_91/cgiirbis_64.exe?P21DBN=KATL&I21DBN=KATL_PRINT&S21FMT=fullw_print&C21COM=F&Z21MFN=15973
5. Якововский М. В. Введение в параллельные методы решения задач: Учебное пособие / Предисл.: В. А. Садовничий. – М.: Издательство Московского университета, 2013. ЭБС «Лань» https://e.lanbook.com/book/73098?category_pk=1557#book_name (число экземпляров неограниченно)
6. Богачёв К.Ю. Основы параллельного программирования: Учебное пособие. – М.: "Лаборатория знаний", 2015. ЭБС «Лань», http://e.lanbook.com/books/element.php?pl1_id=70745 (число экземпляров неограниченно)
7. Федотов И.Е. Модели параллельного программирования. – М.: СОЛОН-Пресс, 2012. ЭБС «Лань», http://e.lanbook.com/books/element.php?pl1_id=13807 (число экземпляров неограниченно)
8. Лацис А.О. Параллельная обработка данных. – М.: Академия, 2010. ЭБС «Академия» <http://www.academia-moscow.ru/catalogue/4831/38049/> (число экземпляров неограниченно)
9. Инструменты параллельного программирования в системах с общей памятью: учебник / К. В. Корняков и [др.]; под ред. В. П. Гергеля ; Нижегородск. гос. ун-т. - 2-е изд., испр. и доп. - М. : Изд-во Моск. ун-та, 2010. (30 экземпляров) http://kleopatra.pnzgu.ru/cgi-bin/irbis64r_91/cgiirbis_64.exe?P21DBN=KATL&I21DBN=KATL_PRINT&S21FMT=fullw_print&C21COM=F&Z21MFN=14058
10. Архитектура и топологии многопроцессорных вычислительных систем / А. Богданов, В. Корхов, В. Мареев, Е. Станкова: курс лекций. НОУ Интуит <http://www.intuit.ru/studies/courses/45/45/info> (число экземпляров неограниченно)
11. Барский А. Архитектура параллельных вычислительных систем: курс лекций. НОУ Интуит <http://www.intuit.ru/studies/courses/80/80/info> (число экземпляров неограниченно)
12. Гергель В. Основы параллельных вычислений: курс лекций. НОУ Интуит <http://www.intuit.ru/studies/courses/1091/293/info> (число экземпляров неограниченно)

б) Интернет-ресурсы ресурсы <http://parallel.ru>, <http://mpi-forum.org>.

в) Программное обеспечение

Все работы выполняются на персональных компьютерах под управлением ОС Ubuntu на языке C++ с использованием библиотеки MPI.

г) Другое материально-техническое обеспечение

При проведении лабораторных занятий используются рабочие места в компьютерном классе, оборудованном локальной сетью и выходом в Internet, имеющиеся в библиотеке учебники. Все лабораторные работы выполняются на персональных компьютерах.

Рабочая программа дисциплины «Вычислительные системы и параллельная обработка данных» составлена в соответствии с требованиями ФГОС ВО по направлению подготовки 01.03.02 «Прикладная математика и информатика», утвержденного приказом Министерства образования и науки Российской Федерации от «10» января 2018 г. № 9

Программу составил:

1. Абрамов И.А., доцент кафедры КТ

Настоящая программа не может быть воспроизведена ни в какой форме без предварительного письменного разрешения кафедры-разработчика программы.

Программа одобрена на заседании кафедры «Компьютерные технологии»

Протокол № 12 от «26» июня 2019 года

Зав. кафедрой «Компьютерные технологии»  В. И. Горбаченко

Программа одобрена методической комиссией факультета вычислительной техники

Протокол № 10 от «03» июня 2019 года

Председатель методической комиссии
Факультета вычислительной техники



(подпись)

Т. В. Глотова
(Ф.И.О.)