	Факультет	Математики, физики и информатики	
	Кафедра	Информатики информационных технологий	
	Направление подготовки	02.03.03 Математическое обеспечение и администрирование информационных систем	
	Направленность (профиль)	Информационные системы и базы данных	
		Параллельное программирование	

Министерство образования и науки Российской Федерации  
 Федеральное государственное бюджетное образовательное учреждение  
 высшего образования  
 «Тульский государственный педагогический университет им.  
 Л.Н. Толстого»  
 ФГБОУ ВО «ТГПУ им. Л.Н. Толстого»

УТВЕРЖДЕНА

на заседании Ученого совета университета  
 протокол № 2 от 11.02.2016 г.

## Рабочая программа дисциплины «Параллельное программирование»


**Трудоемкость: 3 зачетные единицы**

**Квалификация (степень) выпускника: бакалавр**

**Форма обучения: очная**

**Год начала подготовки: 2014**

Рассмотрена на заседании кафедры информатики и информационных технологий  
 протокол № 03 от «18» ноября 2015 г.

Заведующий кафедрой  Якушин А.В.

Одобрена на заседании Ученого совета факультета  
 математики, физики и информатики  
 протокол № 5 от 17.12.2015 г.

Декан факультета  Реброва И.Ю.

## СОДЕРЖАНИЕ

1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы.....	3
2. Место дисциплины в структуре ОПОП бакалавриата.....	3
3. Объем дисциплины и виды учебной работы .....	4
4. Содержание дисциплины, структурированное по темам (разделам) с указанием отведенного на них количества академических или астрономических часов и видов учебных занятий .....	5
5. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине.....	6
6. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине.....	6
6.1. Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы .....	6
6.2. Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания .....	6
6.3. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы.....	7
6.4. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций.....	11
7. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины.....	11
7.1. Основная литература .....	11
7.2. Дополнительная литература .....	11
8. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины .....	11
9. Методические указания для обучающихся по освоению дисциплины .....	12
10. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения и информационных справочных систем .....	14
11. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине .....	14
12. Аннотация рабочей программы дисциплины.....	15
13. Лист регистрации изменений к рабочей программе дисциплины .....	16

# 1. ПЕРЕЧЕНЬ ПЛАНИРУЕМЫХ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ, СООТНЕСЕННЫХ С ПЛАНИРУЕМЫМИ РЕЗУЛЬТАТАМИ ОСВОЕНИЯ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

Достижение планируемых результатов обучения, соотнесенных с общими целями и задачами ОПОП, является целью освоения дисциплины.

Планируемые результаты освоения образовательной программы (код и название компетенции)	Планируемые результаты обучения	Этапы формирования компетенции в процессе освоения образовательной программы
владение разработкой требований и проектирование программного обеспечения (ДПК-2)	<p><b>Выпускник знает:</b> критерии оценки эффективности параллельных программ и их ограничения методы и языковые механизмы конструирования параллельных программ параллельные вычислительные методы</p> <p><b>Умеет:</b> разрабатывать параллельные программы с использованием библиотеки MPI проводить сравнительный анализ последовательных и параллельных программных средств и оценивать их эффективность</p> <p><b>Владеет:</b> навыками конструирования высокопроизводительных вычислительных систем</p>	1 этап из 2 (6 семестр)
Способность использовать основы правовых знаний в различных сферах жизнедеятельности (ОК-4)	<p><b>Выпускник знает:</b> архитектурные принципы реализации параллельной обработки в вычислительных машинах</p> <p><b>Умеет:</b> осуществлять перенос реализованных программных средств на параллельные платформы</p> <p><b>Владеет:</b> навыками настройки высокопроизводительных вычислительных систем</p>	1 этап из 2 (6 семестр)

## 2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП БАКАЛАВРИАТА

Дисциплина «Параллельное программирование» относится к дисциплинам по выбору вариативной части образовательной программы. Изучение данной дисциплины осуществляется в 6 семестре. Изучение данной дисциплины базируется на освоении студентами дисциплин «Архитектура вычислительных систем и компьютерных сетей», «Структуры и алгоритмы компьютерной обработки данных».

К началу изучения дисциплины студенты должны владеть:  
знаниями основных методов, способов и средств обработки данных;  
умениями работать с системами управления базами данных;  
навыками и (или) опытом деятельности методами программирования на алгоритмическом языке C/C++.

Дисциплина «Параллельное программирование» является базовой для дисциплины «Администрирование информационных систем».

**3. ОБЪЕМ ДИСЦИПЛИНЫ И ВИДЫ УЧЕБНОЙ РАБОТЫ**

Очная форма обучения

Вид учебной работы	Объем зачетных единиц / часов по формам обучения
<b>Максимальная учебная нагрузка (всего)</b>	108/3
<b>Контактная работа обучающихся с преподавателем (всего)</b>	22
в том числе:	
лекции	8
лабораторные занятия (включая защиту отчета по лабораторным работам)	
семинарские занятия	
практические занятия	12
контрольные работы	
другие виды контактной работы (КСРС)	2
<b>Самостоятельная работа студента (всего)</b>	86
в том числе:	
внеаудиторная самостоятельная работа по подготовке к лекционным занятиям	22
внеаудиторная самостоятельная работа по подготовке к лабораторным занятиям и защите отчета	
внеаудиторная самостоятельная работа при подготовке к семинарским и/или практическим занятиям	30
подготовка учебного проекта	
подготовка к контрольной работе	
выполнение заданий для самостоятельной работы в системе управления обучением MOODLE	30
выполнение курсового проекта (работы)	
подготовка к зачету	4
подготовка к экзамену	
другие виды самостоятельной работы студента	
Промежуточная аттестация в форме зачета	

**4. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ, СТРУКТУРИРОВАННОЕ ПО ТЕМАМ  
(РАЗДЕЛАМ) С УКАЗАНИЕМ ОТВЕДЕННОГО НА НИХ КОЛИЧЕСТВА  
АКАДЕМИЧЕСКИХ ИЛИ АСТРОНОМИЧЕСКИХ ЧАСОВ И ВИДОВ УЧЕБНЫХ  
ЗАНЯТИЙ**

Наименование тем (разделов).	Количество академических или астрономических часов по видам учебных занятий			
	Занятия лекционного типа	Занятия семинарского типа	Другие виды учебных занятий	Самостоятельная работа обучающихся
Тема 1. Введение в предмет	2	4		20
Тема 2. Вычислительные системы с массовым параллелизмом	2	2		20
Тема 3. Параллельное программирование в MPI.	2	4		22
Тема 4. Реализация параллельных алгоритмов	2	2		20
Контроль самостоятельной работы студентов			2	
Индивидуальные консультации				
Подготовка к зачету				4
Групповые консультации				
<b>ИТОГО</b>	<b>8</b>	<b>12</b>	<b>2</b>	<b>86</b>

**Тема 1. Введение в предмет.** Формальные модели параллельного программирования. Параллельные схемы программ. Информационный базис и схема управления. Асинхронные вычислительные процессы над информационным базисом. Неуправляемые вычислительные процессы. Схема управления параллельной схемой программы. Управляющий автомат и особенности автоматных функций. Эквивалентность вычислительных процессов по произвольному отношению. Детерминизм и отношение "большой параллельности" схем управления.

**Тема 2. Вычислительные системы с массовым параллелизмом.** Определение и примеры. Особенности применения: моделирующие вычислительные системы и системы реального времени. Примеры. Характерные особенности таких систем. Классификация ВС по Флину. Реконфигурируемые системы MIMD архитектуры. Статическая и динамическая реконфигурация. Оценка производительности массивно параллельных ВС. Причины снижения производительности.

**Тема 3. Параллельное программирование в MPI.** Структура библиотеки MPI. Организация обмена сообщениями. Модели вычислений. Кластерные технологии. Организация кластера на базе локальной сети.

**Тема 4. Реализация параллельных алгоритмов.** Параллельные алгоритмы. Общие способы распараллеливания алгоритмов. Организация параллельного исполнения рекурсивных вычислений. Параллельные численные алгоритмы линейной алгебры. Способы разбиения матриц (горизонтальная, вертикальная, блочные схемы). Методы вычисления произведения матриц с использованием разных схем разбиения матриц. Обеспечение предельно допустимого параллелизма. Обращение матриц. Параллельные методы решения систем линейных уравнений. Параллельные численные алгоритмы решения дифференциальных уравнений в частных производных.

## 5. ПЕРЕЧЕНЬ УЧЕБНО-МЕТОДИЧЕСКОГО ОБЕСПЕЧЕНИЯ ДЛЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ

Преподавание дисциплины предполагает использование следующего учебно-методического обеспечения.

Комплекта мультимедийных презентаций для лекционных занятий.

Теоретического курса и информационных приложений, размещенных в электронной образовательной среде MOODLe.

Комплекса заданий для практических занятий, размещенных в электронной образовательной среде MOODLe.

Виды самостоятельной работы обучающихся: выполнение заданий на практические занятия, выполнение индивидуального проектного задания.

При подготовке к занятиям и выполнении самостоятельной работы студентам доступны следующие учебно-методические ресурсы, перечисленные в п.7 рабочей программы, а также электронный учебный ресурс размещенный в среде электронного обучения ТГПУ им. Л.Н. Толстого (<http://moodle.tspu.ru>)

## 6. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ

### 6.1. Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы

Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы представлен в таблице пункта 1 рабочей программы.

Формирование компетенции “Способность использовать основы правовых знаний в различных сферах жизнедеятельности” (ОК-4) осуществляется в течение двух этапов освоения основной образовательной программы.

Первый этап формирования компетенции осуществляется в процессе освоения дисциплины по выбору «Обработка и представление результатов исследований» или «Параллельное программирование».

Второй этап формирования компетенции осуществляется в процессе освоения дисциплины «Правоведение».

Формирование компетенции “Владение разработкой требований и проектирование программного обеспечения ” (ДПК-2) осуществляется в течение двух этапов освоения основной образовательной программы.

Первый этап формирования компетенции осуществляется в процессе освоения дисциплины по выбору «Обработка и представление результатов исследований» или «Параллельное программирование».

Второй этап формирования компетенции осуществляется в процессе освоения дисциплины «Технологии разработки программного обеспечения» и Производственной практики по получению профессиональных умений и опыта профессиональной деятельности.

### 6.2. Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания

Дескриптор компетенций	Показатели оценивания	Критерии оценивания
------------------------	-----------------------	---------------------

Знания	архитектурных принципов реализации параллельной обработки в вычислительных машинах; критерий оценки эффективности параллельных программ и их ограничения; методов и языковых механизмов конструирования параллельных программ; параллельных вычислительных методов.	Отметка «зачтено» выставляется, если студент в целом за семестр набрал от 41 до 100 баллов (с учетом баллов, набранных на промежуточной аттестации (зачете)). Отметка «не зачтено» выставляется, если студент в целом за семестр набрал менее 41 балла (с учетом баллов, набранных на промежуточной аттестации (зачете)).
Умения	разрабатывать параллельные программы с использованием библиотеки MPI; осуществлять перенос реализованных программных средств на параллельные платформы; проводить сравнительный анализ последовательных и параллельных программных средств и оценивать их эффективность	
Навыки и опыт деятельности	конструирования и настройки высокопроизводительных вычислительных систем.	

Критерии оценивания компетенций формируются на основе балльно-рейтинговой системы с помощью всего комплекса методических материалов, определяющих процедуры оценивания знаний, умений, навыков и опыта деятельности, характеризующих данный этап формирования компетенций.

Баллы за промежуточную аттестацию (зачет)	Баллы, набранные студентом в течение семестра	Общая сумма баллов за модуль в семестр	Отметка
20 – 30	21 – 70	41-100	Зачтено
0 – 20	0 – 20	0 – 40	Не зачтено

Оценка «зачтено» ставится, если студент освоил программный материал всех разделов, последователен в изложении программного материала, достаточно последовательно и логически стройно его излагает, умеет увязывать теорию с практикой, успешно прошел текущий контроль успеваемости по дисциплине, продемонстрировал индивидуальные знания, умениями и навыки практической работы.

Оценка «не зачтено» ставится, если студент не знает значительной части программного материала, допускает существенные ошибки, непоследователен в его изложении, не прошел текущий контроль успеваемости, не в полной мере владеет необходимыми знаниями, умениями и навыками при выполнении практических заданий, то есть студент не может продолжить обучение без дополнительной подготовки по соответствующей дисциплине.

### **6.3. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы**

Образцы заданий к практическим занятиям:

1. Написать программу, используя коммуникационные функции (MPI\_Ssend, MPI\_Bsend, MPI\_Rsend, MPI\_Isend, MPI\_Irecv), передающие одномерные и двумерные массивы (вектора и матрицы) между двумя процессорами
2. Провести сравнение по скорости передачи данных в зависимости от применяемых функций и размера передаваемых данных
3. Написать программу, используя коммуникационную функцию (MPI\_Bcast), реализующую алгоритм передачи данных от 0 процесса всем остальным
4. Написать программу, используя коммуникационную функцию (MPI\_Gather), реализующую алгоритм передачи частей массива от всех процессоров на 0.

5. Написать программу, используя коммуникационную функцию (MPI\_Allgather), реализующую алгоритм передачи частей массива от всех процессоров на все процессора.

### Вопросы к зачету

1. Что такое параллельное программирование и суперкомпьютеры. Области в которых может возникать потребность в параллельных вычислениях. Особенности параллельных вычислений.
2. Увеличение производительности при параллельных вычислениях. К каким областям задач может быть применено распараллеливание к каким нет. Закон Амдала.
3. Ускорение параллельного алгоритма.
4. Эффективность параллельного алгоритма.
5. Распараллеливание арифметических выражений. Алгоритм Винограда.
6. Разделение на подзадачи. Установление связей между отдельными подзадачами. Объединение мелких подзадач в большие, законченные счетные единицы (агломерация).
7. Средства поддержки параллельной работы: параллелизм процессов, механизм синхронизации и механизм разделения ресурсов, семафоры, обмен сообщениями.
8. Библиотека MPI. Инициализация, завершение. Базовые функции.
9. Точечные обмены. Синхронный и асинхронный режим.
10. Буферизация сообщений.
11. Отложенные операции. Функции инициализации и завершения операции.
12. Коллективные взаимодействия. Передача сообщений
13. Управление группами и коммуникатором.
14. Коллективные взаимодействия. Прием сообщений
15. Создание и освобождение коммуникатора.
16. Типы данных в MPI.
17. Пакет mpich. Установка и настройка кластера под управлением Linux.
18. Простейшие параллельные алгоритмы и их степень параллелизма.
19. Параллельный алгоритм скалярного умножения векторов и его ускорение по сравнению с последовательным алгоритмом.
20. Параллельный алгоритм умножения матрицы на вектор и его ускорение по сравнению с последовательным алгоритмом.
21. Параллельный алгоритм умножения матрицы на матрицу и его ускорение по сравнению с последовательным алгоритмом.
22. Параллельный алгоритм решения СЛАУ прямым методом Гаусса и его ускорение по сравнению с последовательным алгоритмом.
23. Технологии параллельного программирования.
24. Кластерные технологии.

Индивидуальное проектное задание заключается в разработке параллельной программы:

Каждый проект независимо от темы и задания должен предоставлять пользователю выбор количества узлов, на которых будет запущен процесс вычислений, а также выводить, время, за которое они произошли.

#### 1. Нахождение кратчайших путей для ориентированного графа

Исходной информацией для задачи является взвешенный граф,  $G=(V,R)$  содержащий  $n$  вершин ( $|V|=n$ ), в котором каждому ребру графа приписан неотрицательный вес. Граф будем полагать ориентированным, т.е., если из вершины  $i$  есть ребро в вершину  $j$ , то из этого не



следует наличие ребра из  $j$  в  $i$ . В случае, если вершины все же соединены взаимнообратными ребрами, то веса, приписываемые им, могут не совпадать. Для имеющегося графа  $G$  требуется найти минимальные длины путей между каждой парой вершин графа.

Входные данные начинаются  $N$  количеством вершин от 0 до 10000 и  $M$  количеством ребер в графе от 0 до 10000 через пробел, затем в  $M$  строках записаны номера вершин соединенных  $i$ -ым ребром и его вес от 0 до 10000, где  $i$  – номер строки, начиная счет со второй строки.

Выходные данные должны содержать все кратчайшие пути для всех пар вершин, если такой путь существует для пары. Каждый путь должен быть записан на отдельной строке в виде последовательности номеров ребер через пробел в том порядке, в котором они включены в путь, и к ним добавить последнее после знака « $\Rightarrow$ » число, показывающее вес пути, в начале строки должны быть помещены номера вершин, соединенных этим путем (« $I J : \gg$ »). В случае отсутствия пути между вершинами поставить вместо последовательности вершин прочерк.

Пример входных(input.txt) и выходных(output.txt) данных:

input.txt	output.txt
5 4	1 2 : 1 =1
1 2 1	1 3 : 1 4 = 4
1 3 8	1 4 : 3 = 3
1 4 3	1 5 : -
2 3 2	2 1 : -
3 1 3	2 3 : 4 = 3
4 2 1	2 4 : -
	2 5 : -
	3 1 : 5 = 3
	3 2 : -
	3 4 : -
	3 5 : -
	4 1 : -
	4 2 : 6 = 1
	4 3 : -
	4 5 : -
	5 1 : -
	5 2 : -
	5 3 : -
	5 4 : -

Примечание: в случае если наименьший путь между парой вершин не является единственным, считать кратчайшим тот, который содержит наименьшее количество вершин, если же данное условие не дает единственное решение, вывести любой из них.

## 2. Решение систем линейных уравнений методом алгоритма Гаусса, рекурсивное удвоение Стоуна

Задача: дана расширенная (добавлен вектор-столбец свободных членов) матрица коэффициентов системы линейных уравнений, найти решение системы.

Входной файл в первой строке содержит  $N$  количество строк и  $M$  количество столбцов матрицы через пробел, далее в  $N$  содержится  $M$  элементов (числа от -10000 до 10000) матрицы через пробел. Порядок расположения элементов матриц внутри файла соответствует порядку расположения элементов в самой матрице.

$$-10000 \leq N, M \leq 10000$$

Выходной файл должен содержать значение переменных, совокупность которых является решением данной системы линейных уравнений, через пробел. Порядок расположения значений переменных должен соответствовать порядку номеров их переменных. В случае невозможности нахождения единственного решения вывести в файл сообщение «Error Data». В случае решения системы равного пустому множеству вывести сообщение «No Solutions»

Пример входных(input.txt) и выходных(output.txt) данных:

input.txt	output.txt
1 2 -7 21	-3

input.txt	output.txt
2 3 1 1 5 1 -2 -4	2 3

#### Критерии оценки проектов

Составляющие проекта	Критерии для оценивания	Максимальное количество баллов
Постановка проблемы и ее обоснованность, формулирование целей и задач	<ul style="list-style-type: none"> <li>общественная значимость и актуальность выдвинутых проблем;</li> <li>соответствие темы, цели и задач проекта;</li> <li>разумность масштаба работ.</li> </ul>	10
Содержание проекта/ проектной разработки	<ul style="list-style-type: none"> <li>логичность, взаимосвязь и последовательность этапов проекта;</li> <li>адекватность предлагаемых мероприятий решению поставленных задач;</li> <li>корректность используемых методов работы;</li> <li>четкость определения целевой группы и обоснованность её участия при реализации проекта;</li> <li>соответствие теоретической, эмпирической и проектной частей, их связь с практикой и выбранным видом профессиональной деятельности;</li> <li>соблюдение заявленных временных рамок реализации проекта;</li> <li>самостоятельность и активность участника проекта.</li> </ul>	10
Результат выполнения прикладного проекта	<ul style="list-style-type: none"> <li>соответствие ожиданий от проекта / планируемого результата полученному продукту;</li> <li>степень решения заявленной проблемы;</li> <li>успешность преодоления трудностей в реализации проекта;</li> <li>оценка участников целевой группы;</li> <li>перспективы развития проекта после завершения проекта;</li> <li>возможность тиражирования проекта.</li> </ul>	10
Презентация результатов работы над прикладным проектом	<ul style="list-style-type: none"> <li>ясность, логичность, профессионализм изложения доклада;</li> <li>наглядность и структурированность материала презентации;</li> </ul>	10

	умение корректно использовать профессиональную лексику и понятийно-категориальный аппарат.	
Ответы на вопросы	<ul style="list-style-type: none"> <li>• степень владения темой;</li> <li>• ясность аргументации взглядов студента, презентующего результаты выполнения проекта;</li> <li>четкость и лаконичность ответов на вопросы.</li> </ul>	10

#### **6.4. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций**

Описание балльно-рейтинговой системы по дисциплине.

Итоговая рейтинговая оценка по дисциплине складывается из следующих составляющих:

- 1) В течение семестра за выполнение заданий по курсу студент может максимально получить 70 баллов;
- 2) Обязательной формой текущей аттестации знаний является выполнение индивидуального проектного задания 20 баллов.
- 3) На зачёте ответ студента может быть максимально оценен в 30 баллов.

При этом, для получения положительной итоговой оценки на зачете необходимо получить не менее 60% по каждой составляющей и выполнить все задания для практических занятий. Шкала перевода баллов в оценку: до 40 - «не зачтено»; 41 - 100 - «зачтено».

№ п/п	Критерии оценивания	Максимальное количество баллов	Баллы, полученные студентом
1.	Выполнение заданий:	80	
1.1.	Практические занятия	60	
1.2.	Индивидуальное проектное задание	20	
3.	Зачет	20	
	ИТОГО:	100	

### **7. ПЕРЕЧЕНЬ ОСНОВНОЙ И ДОПОЛНИТЕЛЬНОЙ УЧЕБНОЙ ЛИТЕРАТУРЫ, НЕОБХОДИМОЙ ДЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ**

#### **7.1. Основная литература**

1. Основы параллельного программирования [Электронный ресурс] : учебное пособие / К. Ю. Богачёв. - М : [б. и.], 2013. - 344 с. - ISBN 978-5-9963-0939-9 : Б. ц.  
URL:<http://biblioclub.ru/index.php?page=book&id=214157>
2. Мартынюк, Ю. М. Методы программирования [Текст] : учебное пособие / Ю. М. Мартынюк, С. С. Гербут, В. С. Ванькова ; рец.: Е. Г. Торина, Е. А. Снижко ; ФГБОУ ВПО "Тульский государственный педагогический университет им. Л. Н. Толстого". - Тула : Изд-во ТГПУ им. Л. Н. Толстого, 2013. - 70 с.

#### **7.2. Дополнительная литература**

1. Основы информатики и начала программирования [Текст] : учебное пособие / В. Г. Куперман, Е. Г. Торина, Изд. 2-е, перераб. и доп. - Тула : Изд-во ТГПУ им. Л. Н. Толстого, 1997. - 264 с. - ISBN 5879541169

2. Объектно-ориентированное программирование на С++ [Текст] / Айра Пол. - 2-е изд. - М. : Бином, 1999. - 462 с. : ил. - ISBN 5798901408
3. Технологии параллельного программирования [Текст] : учебное пособие для студентов вузов / С. А. Лупин, М. А. Посыпкин. - М : Форум, 2008. - 208 с. - ISBN 9785819903360
4. С++.Объектно-ориентированное программирование [Текст] : учебное пособие для студентов вузов / В. В. Лаптев, А. В. Морозов, А. В. Бокова. - С П б. : Питер, 2008. - 464 с. : ил. - ISBN 9785911802004

#### **8. ПЕРЕЧЕНЬ РЕСУРСОВ ИНФОРМАЦИОННО-ТЕЛЕКОММУНИКАЦИОННОЙ СЕТИ «ИНТЕРНЕТ», НЕОБХОДИМЫХ ДЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ**

1. Math-Net.Ru [Электронный ресурс] : общероссийский математический портал / Математический институт им. В. А. Стеклова РАН ; Российская академия наук, Отделение математических наук. - М. : [б. и.], 2010. - Загл. с титул. экрана. - Б. ц.  
URL: <http://www.mathnet.ru>
2. ИКТ [Электронный ресурс] : федеральный образовательный портал / ФГАУ ГНИИ ИТТ "Информика". - М. : [б. и.], 2003. - Загл. с титул. экрана. - Б. ц.  
URL: <http://www.ict.edu.ru>
3. Университетская библиотека Online [Электронный ресурс] : электронная библиотечная система / ООО "Директ-Медиа" . - М. : [б. и.], 2001. - Загл. с титул. экрана. - Б. ц.  
URL: [www.biblioclub.ru](http://www.biblioclub.ru)
4. Универсальные базы данных East View [Электронный ресурс] : информационный ресурс / East View Information Services. - М. : [б. и.], 2012. - Загл. с титул. экрана. - Б. ц.  
URL: [www.ebiblioteka.ru](http://www.ebiblioteka.ru)
5. Научная электронная библиотека eLIBRARY.RU [Электронный ресурс] : информационный портал / ООО "РУНЭБ" ; Санкт-Петербургский государственный университет. - М. : [б. и.], 2005. - Загл. с титул. экрана. - Б. ц.  
URL: [www.eLibrary.ru](http://www.eLibrary.ru)

#### **9. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ**

Приступая к изучению новой учебной дисциплины, студенты должны ознакомиться с учебной программой, учебной, научной и методической литературой, имеющейся в библиотеке университета, встретиться с преподавателем, ведущим дисциплину, получить в библиотеке рекомендованные учебники и учебно-методические пособия, осуществить запись на соответствующий курс в среде электронного обучения университета.

Глубина усвоения дисциплины зависит от активной и систематической работы студента на лекциях и практических занятиях, а также в ходе самостоятельной работы, по изучению рекомендованной литературы.

На лекциях важно сосредоточить внимание на ее содержании. Это поможет лучше воспринимать учебный материал и уяснить взаимосвязь проблем по всей дисциплине. Основное содержание лекции целесообразнее записывать в тетради в виде ключевых фраз, понятий, тезисов, обобщений, схем, опорных выводов. Необходимо обращать внимание на термины, формулировки, раскрывающие содержание тех или иных явлений и процессов, научные выводы и практические рекомендации. Желательно оставлять в конспектах поля, на которых делать пометки из рекомендованной литературы, дополняющей материал прослушанной лекции, а также подчеркивающие особую важность тех или иных теоретических положений. С целью уяснения теоретических положений, разрешения спорных ситуаций необходимо задавать преподавателю уточняющие вопросы. Для закрепления содержания лекции

в памяти, необходимо во время самостоятельной работы внимательно прочесть свой конспект и дополнить его записями из учебников и рекомендованной литературы. Конспектирование читаемых лекций и их последующая доработка способствует более глубокому усвоению знаний, и поэтому являются важной формой учебной деятельности студентов.

Прочное усвоение и долговременное закрепление учебного материала невозможно без продуманной самостоятельной работы. Такая работа требует от студента значительных усилий, творчества и высокой организованности. В ходе самостоятельной работы студенты выполняют следующие задачи: дорабатывают лекции, изучают рекомендованную литературу, готовятся к практическим занятиям, к коллоквиуму, контрольным работам по отдельным темам дисциплины. При этом эффективность учебной деятельности студента во многом зависит от того, как он распорядился выделенным для самостоятельной работы бюджетом времени.

Результатом самостоятельной работы является прочное усвоение материалов по предмету согласно программы дисциплины. В итоге этой работы формируются профессиональные умения и компетенции, развивается творческий подход к решению возникших в ходе учебной деятельности проблемных задач, появляется самостоятельности мышления.

Целью практических занятий по данной дисциплине является закрепление теоретических знаний, полученных при изучении дисциплины.

При подготовке к практическому занятию целесообразно выполнить следующие рекомендации: изучить основную литературу; ознакомиться с дополнительной литературой, новыми публикациями в периодических изданиях: журналах, газетах и т. д.; при необходимости доработать конспект лекций. При этом учесть рекомендации преподавателя и требования учебной программы.

При выполнении практических занятий основным методом обучения является самостоятельная работа студента под управлением преподавателя. На них пополняются теоретические знания студентов, их умение творчески мыслить, анализировать, обобщать изученный материал, проверяется отношение студентов к будущей профессиональной деятельности.

Оценка выполненной работы осуществляется преподавателем комплексно: по результатам выполнения заданий, устному сообщению и оформлению работы. После подведения итогов занятия студент обязан устранить недостатки, отмеченные преподавателем при оценке его работы.

Преподавание дисциплины должно включать в себя следующие образовательные технологии:

- 1) Проведение лекций с использованием презентаций на основе мультимедийных технологий;
- 2) Обеспечение студентов сопутствующими материалами, размещенными в среде Moodle; Примерная тематика практических занятий по дисциплине.

Полные варианты практических занятий размещены в в системе управления обучением MOODLE.

№	Наименование практических занятий	Объем в часах
1	Кластерные технологии	2
2	Основы MPI	2
3	Простейшие параллельные алгоритмы	4
4	Распараллеливание вычислений.	4
	Итого	12

#### Типовые задания для самостоятельной работы по дисциплине

1. Нахождение кратчайших путей для ориентированного графа
2. Нахождение кратчайших путей для неориентированного графа
3. Нахождение минимального остового дерева
4. Сортировка Шелла
5. Быстрая сортировка

6. Умножение матриц делением на полосы (строки или столбцы) — ленточная схема
7. Умножение матриц делением на блоки (подматрицы) — блочное представление, алгоритм Фокса
8. Умножение матриц делением на блоки (подматрицы) — блочное представление, алгоритм Кэннона
9. Решение систем линейных уравнений методом алгоритма Гаусса, рекурсивное удвоение Стоуна
10. Последовательная и каскадная суммы
11. Вычисление определенного интеграла: метод Гаусса
12. Вычислить сумму ряда

**10. ПЕРЕЧЕНЬ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, ИСПОЛЬЗУЕМЫХ ПРИ  
ОСУЩЕСТВЛЕНИИ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ,  
ВКЛЮЧАЯ ПЕРЕЧЕНЬ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ И  
ИНФОРМАЦИОННЫХ СПРАВОЧНЫХ СИСТЕМ**

1. Подписка Microsoft DreamSpark Premium - Сублицензионный договор № S-2042626/M18 от 04.06.2013:
  - 1.1. Средства для разработки и проектирования Visual Studio 2008, 2010, 2012 и 2013 Professional Editions;
  - 1.2. Интегрированная среда разработки Visual Studio Express;
  - 1.3. Операционная система Windows Server 2008 Standard Edition 32-bit;
  - 1.4. Операционная система Windows 8.1 Pro;
  - 1.5. Отдельные программы из Office 2007, Office 2010, Office 2013;
2. Операционная система Microsoft Windows XP Professional Russian – Лицензия № 16698685 от 08.08.2003 г.;
3. Программное обеспечение Microsoft Office XP Professional Win32 Russian– Лицензия № 16698685 от 08.08.2003 г.;
4. Веб-браузеры.
5. Доступ студентов через личные кабинеты к электронным библиотечным системам.
6. Возможность работы студентов на удаленном рабочем столе кафедры информатики и информационных технологий.

**11. ОПИСАНИЕ МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЙ БАЗЫ, НЕОБХОДИМОЙ ДЛЯ  
ОСУЩЕСТВЛЕНИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ**

- компьютерный класс для проведения лабораторных занятий по дисциплине, оснащённый компьютерами с процессорами классов Pentium или Core Duo (количество компьютеров – не менее 10 укомплектованных компьютерами рабочих мест);
- видеопроектор и в качестве средства поддержки лекционных занятий;
- интерактивная доска в качестве средства поддержки лекционных занятий;
- Интернет-доступ, позволяющий осуществлять подбор материалов для выполнения заданий, подготовки информационного проекта, научных сообщений, реферата;
- аудитории для самостоятельной работы студентов, оснащенные компьютерной техникой, имеющей доступ к информационно-телекоммуникационной сети «Интернет», электронной информационно-образовательной среде ТГПУ им. Л.Н. Толстого, внутривузовскому сетевому окружению;
- наличие прав доступа к перечисленному выше программному обеспечению

## 12. АННОТАЦИЯ РАБОЧЕЙ ПРОГРАММЫ ДИСЦИПЛИНЫ.

1. Планируемые результаты обучения при освоении дисциплины, соотнесенные с планируемыми результатами освоения образовательной программы.

В результате освоения дисциплины у студента должна быть сформирована следующая компетенция: владение разработкой требований и проектирование программного обеспечения (ДПК-2).

В результате освоения дисциплины студент должен приобрести:

критерий оценки эффективности параллельных программ и их ограничения; методов и языковых механизмов конструирования параллельных программ; параллельных вычислительных методов.

умения разрабатывать параллельные программы с использованием библиотеки MPI; проводить сравнительный анализ последовательных и параллельных программных средств и оценивать их эффективность

навыки конструирования высокопроизводительных вычислительных систем.

В результате освоения дисциплины у студента должна быть сформирована следующая компетенция: способность использовать основы правовых знаний в различных сферах жизнедеятельности (ОК-4).

В результате освоения дисциплины студент должен приобрести:

знания архитектурных принципов реализации параллельной обработки в вычислительных машинах;

умения осуществлять перенос реализованных программных средств на параллельные платформы;

навыки настройки высокопроизводительных вычислительных систем.

### 2. Место дисциплины в структуре ОПОП.

Дисциплина «Параллельное программирование» относится к дисциплинам по выбору вариативной части образовательной программы. Изучение данной дисциплины осуществляется в 6 семестре. Изучение данной дисциплины базируется на освоении студентами дисциплин «Архитектура вычислительных систем и компьютерных сетей», «Структуры и алгоритмы компьютерной обработки данных».

К началу изучения дисциплины студенты должны владеть:

знаниями основных методов, способов и средств обработки данных;

умениями работать с системами управления базами данных;

навыками и (или) опытом деятельности методами программирования на алгоритмическом языке C/C++.

Дисциплина «Параллельное программирование» является базовой для дисциплины «Администрирование информационных систем».

3. Объем дисциплины: 3 зачетные единицы.

4. Образовательный процесс осуществляется на русском языке.

5. Разработчик: Якушин А.В., к.п.н., доцент, зав. кафедрой ИиИТ.

### 13. ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ К РАБОЧЕЙ ПРОГРАММЕ ДИСЦИПЛИНЫ

Внесены изменения в п.7 «Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины».

Обновлен п.10 «Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень лицензионного программного обеспечения и информационных справочных систем» на основании действующих лицензионных соглашений.

Заведующий кафедрой ИиИТ



А.В. Якушин

«29» августа 2016 г.



Программа составлена в соответствии с требованиями ФГОС ВО.

**Разработчики:**

Фамилия, имя, отчество	Учёная степень	Учёное звание	Должность	Дата разработки	Подпись
Якушин Алексей Валериевич	к.п.н.	Доцент	Зав. кафедрой информатики и информационных технологий	17.11.2015	