

Документ подписан простой электронной подписью  
Информация о владельце:  
ФИО: Галунин Сергей Александрович  
Должность: проректор по учебной работе  
Дата подписания: 18.07.2023 17:12:32  
Уникальный программный ключ:  
08ef34338325bdb0ac5a47baa5472ce36cc3fc3b

Приложение к ОПОП  
«Системы искусственного интел-  
лекта»



**СПбГЭТУ «ЛЭТИ»**  
ПЕРВЫЙ ЭЛЕКТРОТЕХНИЧЕСКИЙ

МИНОБРНАУКИ РОССИИ

федеральное государственное автономное образовательное учреждение высшего образования  
**«Санкт-Петербургский государственный электротехнический университет  
«ЛЭТИ» им. В.И.Ульянова (Ленина)»**  
**(СПбГЭТУ «ЛЭТИ»)**

---

**РАБОЧАЯ ПРОГРАММА**

ДИСЦИПЛИНЫ

**«ПАРАЛЛЕЛЬНЫЕ АЛГОРИТМЫ»**

для подготовки бакалавров

по направлению

01.03.02 «Прикладная математика и информатика»

по профилю

**«Системы искусственного интеллекта»**

Санкт-Петербург

2023

## ЛИСТ СОГЛАСОВАНИЯ

Разработчики:

к.т.н., ассистент Сергеева Е.И.

Рабочая программа рассмотрена и одобрена на заседании кафедры МОЭВМ  
20.09.2022, протокол № 7

Рабочая программа рассмотрена и одобрена учебно-методической комиссией  
ФКТИ, 29.09.2022, протокол № 7

Согласовано в ИС ИОТ

Начальник ОМОЛА Загороднюк О.В.

## 1 СТРУКТУРА ДИСЦИПЛИНЫ

Обеспечивающий факультет	ФКТИ
Обеспечивающая кафедра	МОЭВМ
Общая трудоемкость (ЗЕТ)	4
Курс	3
Семестр	5
<b>Виды занятий</b>	
Лекции (академ. часов)	34
Практические занятия (академ. часов)	17
Иная контактная работа (академ. часов)	1
Все контактные часы (академ. часов)	52
Самостоятельная работа, включая часы на контроль (академ. часов)	92
Всего (академ. часов)	144
<b>Вид промежуточной аттестации</b>	
Дифф. зачет (курс)	3

## **2 АННОТАЦИЯ ДИСЦИПЛИНЫ**

### **«ПАРАЛЛЕЛЬНЫЕ АЛГОРИТМЫ»**

Параллельные алгоритмы весьма важны ввиду постоянного совершенствования многопроцессорных систем и увеличения числа ядер в современных процессорах. В учебном курсе излагается учебный материал, достаточный для успешного начала работ в области параллельного программирования. Для этого в пособии дается краткая характеристика принципов построения параллельных вычислительных систем, рассматриваются математические модели параллельных алгоритмов и программ для анализа эффективности параллельных вычислений, приводятся примеры конкретных параллельных методов для решения типовых задач вычислительной математики. Особое внимание уделено разработке параллельных алгоритмов с учетом архитектуры параллельного вычислителя.

### **SUBJECT SUMMARY**

#### **«PARALLEL ALGORITHMS»**

Parallel algorithms are very important considering the continuous improvement of multi-processor systems and increasing number of cores in modern processors. This tutorial presents course material sufficient for a good start in the field of parallel programming. The manual provides a brief description of the principles of constructing parallel computing systems, Mathematical models of parallel algorithms and software for performance analysis of parallel computing. Also examples of specific parallel methods for solving typical problems of computational mathematics are given. Special attention is paid to the development of parallel algorithms considering the architecture of the parallel computing system.

## 3 ОБЩИЕ ПОЛОЖЕНИЯ

### 3.1 Цели и задачи дисциплины

1. Целью дисциплины является приобретение теоретических знаний и практических навыков практического применения классификации и архитектуры параллельных вычислительных систем, средств современных операционных систем для поддержки параллелизма, моделей параллельного программирования, методов и средств оценки эффективности параллельных программ.
2. Задачи дисциплины: формирование умения осуществлять выбор архитектуры параллельной вычислительной системы в зависимости от типа решаемой задачи обработки данных; умения проектировать эффективные параллельные алгоритмы; знания работы в программной среде параллельного программирования и создания эффективных масштабируемых параллельных программ.
3. Знание классификации и архитектуры параллельных вычислительных систем; средств современных операционных систем для поддержки параллелизма; моделей параллельного программирования;– методов и средств оценки эффективности параллельных программ.
4. Формирование: умения осуществлять выбор архитектуры параллельной вычислительной системы в зависимости от типа решаемой задачи обработки данных;– умения проектировать эффективные параллельные алгоритмы;– знания работы в программной среде параллельного программирования и создания эффективных масштабируемых параллельных программ.
5. Освоение навыков работы в средах параллельного программирования;– навыков разработки и отладки параллельных алгоритмов и программ, с учетом модели параллельного вычислителя.

### **3.2 Место дисциплины в структуре ОПОП**

Дисциплина изучается на основе ранее освоенных дисциплин учебного плана:

1. «Информатика»
2. «Программирование»
3. «Информационные технологии»
4. «Алгоритмы и структуры данных»
5. «Организация ЭВМ и систем»
6. «Операционные системы»

и обеспечивает изучение последующих дисциплин:

1. «Основы промышленной разработки программного обеспечения»
2. «Тестирование программного обеспечения»
3. «Введение в автономные интеллектуальные системы»

### 3.3 Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы

В результате освоения образовательной программы обучающийся должен достичь следующие результаты обучения по дисциплине:

<b>Код компетенции/ индикатора компетенции</b>	<b>Наименование компетенции/индикатора компетенции</b>
ПК-0	Способен разрабатывать информационные модели и применять их для решения задач профессиональной деятельности
<i>ПК-0.3</i>	<i>Применяет информационные модели для решения задач профессиональной деятельности</i>

## 4 СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

### 4.1 Содержание разделов дисциплины

#### 4.1.1 Наименование тем и часы на все виды нагрузки

№ п/п	Наименование темы дисциплины	Лек, ач	Пр, ач	ИКР, ач	СР, ач
1	Введение	1			
2	Тема 1. Основы параллельного программирования	4	3		14
3	Тема 2. Процессы операционной системы	8	4		14
4	Тема 3. Потоки операционной системы	8	4		14
5	Тема 4. Технология параллельного программирования OpenMP	6	3		14
6	Тема 5. Примеры реализации базовых параллельных алгоритмов	6	3		15
7	Заключение	1		1	21
	Итого, ач	34	17	1	92
	Из них ач на контроль	0	0	0	0
	Общая трудоемкость освоения, ач/зе	144/4			

#### 4.1.2 Содержание

№ п/п	Наименование темы дисциплины	Содержание
1	Введение	Методы и принципы построения параллельных алгоритмов и реализации параллельных программ являются общими и могут быть использованы при работе с различными вычислительными системами. В курсе рассматриваются вычислительные системы с общей памятью.



№ п/п	Наименование темы дисциплины	Содержание
2	Тема 1. Основы параллельного программирования	Краткое введение в архитектуры вычислительных систем: системы с общей памятью и системы с распределенной памятью. Цели параллельного программирования: «Уменьшение времени решения задачи»; «Решение задач с большим объемом исходных данных»; «Получение корректного результата с большей вероятностью». Теоретические основы параллельных вычислений. Этапы разработки параллельного алгоритма: поиск параллелизма, декомпозиция задачи, анализ зависимостей. Этапы реализации параллельной программы: распределение задач по процессорам, организация взаимодействия подзадач, измерение и анализ эффективности параллельной программы. Средства и методы анализа эффективности параллельных алгоритмов: коэффициент ускорения, коэффициент эффективности, коэффициент накладных расходов, показатель равномерности загрузки параллельных ветвей.
3	Тема 2. Процессы операционной системы	Введение в ОС UNIX. Краткое введение в архитектуру операционной системы UNIX. Понятие процесса. Атрибуты процесса. Методы планирования процессов. Жизненный цикл процесса. Системные вызовы управление процессами. Средства межпроцессного взаимодействия: сигналы, именованные и неименованные каналы, разделяемая память, семафоры, очереди сообщений, сокеты. Системные вызовы для организации межпроцессного взаимодействия.
4	Тема 3. Поток операционной системы	Потоки PThreads. Атрибуты потоков. Жизненный цикл потока. Системные вызовы управления потоками. Средства синхронизации потоков.
5	Тема 4. Технология параллельного программирования OpenMP	Введение в технологию OpenMP. Модель параллелизма. Составные части программного пакета OpenMP: переменные окружения, библиотечные функции, директивы. Основные директивы. Преимущества и недостатки технология OpenMP.
6	Тема 5. Примеры реализации базовых параллельных алгоритмов	Алгоритмы обработки матриц. Алгоритмы сортировки. Алгоритмы на графах.
7	Заключение	Основные выводы по курсу. Перспективы развития.

## 4.2 Перечень лабораторных работ

Лабораторные работы не предусмотрены.

## 4.3 Перечень практических занятий

<b>Наименование практических занятий</b>	<b>Количество ауд. часов</b>
1. Реализация заданной иерархии вычислительных процессов	2
2. Взаимодействие параллельных процессов по средствам программных каналов	3
3. Передача данных между параллельными процессами через разделяемую память	2
4. Синхронизация доступа к разделяемой памяти параллельными процессами	2
5. Реализация параллельных вычислений на основе потоков	4
6. Обмен данными между процессами по средствам коммуникационной сети	2
7. Реализация параллельных вычислений на основе OpenMP	2
Итого	17

#### **4.4 Курсовое проектирование**

Курсовая работа (проект) не предусмотрены.

#### **4.5 Реферат**

Реферат не предусмотрен.

#### **4.6 Индивидуальное домашнее задание**

Индивидуальное домашнее задание не предусмотрено.

#### **4.7 Доклад**

В начале семестра студенты получают темы для подготовки презентации и доклада.

Цель доклада - описать технологию, метод, алгоритм, о котором идет речь в предложенной теме; выявить достоинства и недостатки, привести примеры реализации в системах параллельного программирования.

Доклад представляется презентацией с устным сопровождением. Презентация должна содержать не менее 3 и не более 5 слайдов. Длительность доклада не должна превышать 12 минут включая вопросы слушателей. Доклады представ-

ляются и защищаются студентами на практических занятиях в конце семестра.

Примерные темы для доклада:

1. Технология OpenMP
2. Технология MPI
3. Парадигма вычислений MapReduce
4. Параллельный алгоритм сортировки слиянием
5. Параллельный алгоритм пузырьковой сортировки

Выступление оценивается исходя из экспертной оценки преподавателя на основании следующих критериев: полнота освещения темы, информативность представленной презентации, ответы на вопросы.

*Неудовлетворительно* - Тема не освещена, презентация неинформативна, студент испытывает затруднения при ответах на вопросы.

*Удовлетворительно* - Студент в целом раскрыл тему, но презентация малоинформативна, студент неполно отвечает на вопросы.

*Хорошо* - Студент хорошо раскрыл тему, но при ответе на отдельные вопросы испытывает затруднения. Презентация выполнена качественно.

*Отлично* - Студент полно осветил тему доклада, представил качественную презентацию, свободно и полно отвечает на вопросы.

#### **4.8 Кейс**

Кейс не предусмотрен.

#### **4.9 Организация и учебно-методическое обеспечение самостоятельной работы**

Изучение дисциплины сопровождается самостоятельной работой студентов с рекомендованными преподавателем литературными источниками и ин-

формационными ресурсами сети Интернет.

Планирование времени для изучения дисциплины осуществляется на весь период обучения, предусматривая при этом регулярное повторение пройденного материала. Обучающимся, в рамках внеаудиторной самостоятельной работы, необходимо регулярно дополнять сведениями из литературных источников материал, законспектированный на лекциях. При этом на основе изучения рекомендованной литературы целесообразно составить конспект основных положений, терминов и определений, необходимых для освоения разделов учебной дисциплины.

Особое место уделяется консультированию, как одной из форм обучения и контроля самостоятельной работы. Консультирование предполагает особым образом организованное взаимодействие между преподавателем и студентами, при этом предполагается, что консультант либо знает готовое решение, которое он может предписать консультируемому, либо он владеет способами деятельности, которые указывают путь решения проблемы.

<b>Текущая СРС</b>	<b>Примерная трудоемкость, ач</b>
Работа с лекционным материалом, с учебной литературой	17
Опережающая самостоятельная работа (изучение нового материала до его изложения на занятиях)	0
Самостоятельное изучение разделов дисциплины	0
Выполнение домашних заданий, домашних контрольных работ	0
Подготовка к лабораторным работам, к практическим и семинарским занятиям	17
Подготовка к контрольным работам, коллоквиумам	0
Выполнение расчетно-графических работ	0
Выполнение курсового проекта или курсовой работы	0
Поиск, изучение и презентация информации по заданной проблеме, анализ научных публикаций по заданной теме	0
Работа над междисциплинарным проектом	0
Анализ данных по заданной теме, выполнение расчетов, составление схем и моделей, на основе собранных данных	37
Подготовка к зачету, дифференцированному зачету, экзамену	21
<b>ИТОГО СРС</b>	<b>92</b>

## 5 Учебно-методическое обеспечение дисциплины

### 5.1 Перечень основной и дополнительной литературы, необходимой для освоения дисциплины

№ п/п	Название, библиографическое описание	К-во экз. в библ.
Основная литература		
1	Мараховский, Вячеслав Борисович. Моделирование параллельных процессов. Сети Петри [Текст] : курс для систем. архитекторов, программистов, систем. аналитиков, проектировщиков слож. систем управления / В. Б. Мараховский, Л. Я. Розенблюм, А. В. Яковлев, 2014. -398 с.	30
2	Параллельные вычисления в локальных сетях [Текст] : метод. указания к лаб. работам по дисциплине "Параллельные алгоритмы и системы" / Санкт-Петербургский государственный электротехнический университет им. В.И. Ульянова (Ленина) "ЛЭТИ", 2005. -44 с.	83
3	Энтони У. Параллельное программирование на C++ в действии. Практика разработки многопоточных программ [Электронный ресурс], 2012. -672 с.	неогр
Дополнительная литература		
1	Робачевский, Андрей М. Операционная система UNIX [Текст] : Учеб. пособие для вузов / А.М.Робачевский, 2002. -IX, 514 с.	15
2	Немнюгин, Сергей А. Параллельное программирование для многопроцессорных вычислительных систем [Текст] : монография / С.А.Немнюгин, О.Л.Стефик, 2002. -396 с.	10
3	Макконелл Дж. Анализ алгоритмов. Вводный курс [Текст] : учебное пособие / Дж. Макконелл; пер. с англ. С.К. Ландо, 2002. -302 с.	9
4	Седжвик, Роберт. Фундаментальные алгоритмы на С [Текст] : [пер. с англ.]. Ч. 1-5 : Анализ структуры данных. Сортировка. Поиск. Алгоритмы на графах / Р. Седжвик, 2003. -1127 с.	40
5	Таненбаум, Эндрю. Современные операционные системы [Текст] : монография / Э. Таненбаум; [Пер. на рус. яз. А. Леонтьев], 2002. -1037 с.	79

### 5.2 Перечень ресурсов информационно-телекоммуникационной сети «Интернет», используемых при освоении дисциплины

№ п/п	Электронный адрес
1	Лаборатория Параллельных информационных технологий / Научно-исследовательского вычислительного центра Московского государственного университета имени М.В.Ломоносова <a href="https://parallel.ru/about">https://parallel.ru/about</a>

### **5.3 Адрес сайта курса**

Адрес сайта курса: <https://vec.etu.ru/moodle/course/view.php?id=10625>

## 6 Критерии оценивания и оценочные материалы

### 6.1 Критерии оценивания

Для дисциплины «Параллельные алгоритмы» предусмотрены следующие формы промежуточной аттестации: зачет с оценкой.

#### Зачет с оценкой

<b>Оценка</b>	<b>Описание</b>
Неудовлетворительно	Курс не освоен. Студент испытывает серьезные трудности при ответе на ключевые вопросы дисциплины
Удовлетворительно	Студент в целом овладел курсом, но некоторые разделы освоены на уровне определений и формулировок
Хорошо	Студент овладел курсом, но в отдельных вопросах испытывает затруднения. Умеет решать задачи
Отлично	Студент демонстрирует полное овладение курсом, способен применять полученные знания при решении конкретных задач.

## Особенности допуска

Студенты, получившие отметку ”зачтено” за 7 практических работ и оценку за доклад не ниже ”удовлетворительно”, допускаются к промежуточной аттестации. Оценка за промежуточную аттестацию по выбору студента может быть выставлена на основании оценки за доклад, либо студент проходит промежуточную аттестацию в форме тестирования.

## 6.2 Оценочные материалы для проведения текущего контроля и промежуточной аттестации обучающихся по дисциплине

### Вопросы к дифф.зачету

№ п/п	Описание
1	Этапы разработки параллельного алгоритма: поиск параллелизма, декомпозиция задачи, анализ зависимостей
2	Средства и методы анализа эффективности параллельных алгоритмов: коэффициент ускорения
3	Понятие процесса. Атрибуты процесса.
4	Средства межпроцессного взаимодействия: сигналы
5	Потоки PThreads
6	Средства синхронизации потоков
7	Составные части программного пакета OpenMP
8	Преимущества и недостатки технологии OpenMP
9	Алгоритмы обработки матриц
10	Алгоритмы на графах

### Вариант теста

Тесты предусматривают вопросы с возможностью выбора только одного варианта правильного ответа, либо правильных ответов может быть несколько.

#### 1. Возможно ли параллельное выполнение программного кода на вычислительной машине с одним физическим процессором (ядром)

- а. Нет
- б. Да, на всех процессорах с SIMD набором инструкций
- в. Да, только на процессорах с 64-битной архитектурой



**2. Переключения контекста на процессоре выполняется быстрее для процесса или потока?**

- а. Для процесса
- б. Для потока
- в. Одинаково

**3. Что такое livelock?**

- а. Взаимная блокировка нескольких вычислительных потоков
- б. Метод синхронизации потоков
- в. Ситуация при которой несколько потоков бесконечно захватывают и отпускают примитив синхронизации

**Образцы задач (заданий) для контрольных (проверочных) работ**

**Примеры вопросов при защите практических работ на коллоквиумах**

1. Опишите процессы операционной системы.
2. Какими свойствами и атрибутами обладают процессы?
3. Что такое жизненный цикл процесса?
4. Опишите средства межпроцессного взаимодействия.
5. Как работают именованные и неименованные каналы?
6. Что такое жизненный цикл потока?
7. Охарактеризуйте технологии параллельного программирования OpenMP.

Весь комплект контрольно-измерительных материалов для проверки сформированности компетенции (индикатора компетенции) размещен в закрытой

части по адресу, указанному в п. 5.3

### 6.3 График текущего контроля успеваемости

Неделя	Темы занятий	Вид контроля
1	Тема 1. Основы параллельного программирования	
2		
3		Коллоквиум
4	Тема 1. Основы параллельного программирования Тема 2. Процессы операционной системы	
5		Коллоквиум
6	Тема 2. Процессы операционной системы Тема 3. Потоки операционной системы	
7		Коллоквиум
8	Тема 3. Потоки операционной системы	
9		Коллоквиум
10	Тема 3. Потоки операционной системы Тема 4. Технология параллельного программирования OpenMP	
11		Коллоквиум
12	Тема 4. Технология параллельного программирования OpenMP	
13		Коллоквиум
14	Тема 5. Примеры реализации базовых параллельных алгоритмов	
15		Коллоквиум
16	Заключение	
17		Доклад / Презентация

### 6.4 Методика текущего контроля

#### на лекционных занятиях

Текущий контроль включает в себя:

– контроль посещаемости (не менее 80% занятий).

#### на практических занятиях

В процессе обучения по дисциплине студент обязан выполнить и успешно защитить на коллоквиумах 7 практических работ. Под выполнением практических работ подразумевается подготовка к работе, выполнение задания, подготовка отчета и его защита. Отчет оформляется после выполнения задания и представляется преподавателю на проверку. После проверки отчет либо возвращается (при наличии замечаний) на доработку, либо допускается к защите.

Практические работы защищаются студентами индивидуально, во время

коллоквиумов. На защите работы студент должен показать: понимание постановки задачи, подхода к ее решению, умение объяснять ход решения, выбор тех или иных методик решения задачи. Преподаватель задает вопросы, позволяющие определить глубину понимания теоретического материала, который лежит в основе решения задачи практической работы, а также самостоятельность ее выполнения.

Текущий контроль включает в себя контроль выполнения практической работы и сдачи в срок отчета.

Критерии оценивания: «не зачтено» - ставится, если основное содержание материала работы не раскрыто, не даны ответы на вопросы преподавателя, допущены грубые ошибки в определении понятий и в использовании терминологии; «зачтено» ставится, если продемонстрировано усвоение основного содержания материала, работа выполнена полностью, самостоятельно и оформлена в соответствии с требованиями.

#### **самостоятельной работы студентов.**

Контроль самостоятельной работы студентов осуществляется на лекциях, лабораторных и практических занятиях по методикам, описанным выше.

## 7 Описание информационных технологий и материально-технической базы

Тип занятий	Тип помещения	Требования к помещению	Требования к программному обеспечению
Лекция	Лекционная аудитория	Количество посадочных мест – в соответствии с контингентом, рабочее место преподавателя, IBM-совместимый компьютер Pentium или выше, проектор, экран, меловая или маркерная доска	1) Linux Альт Образование 10 и выше; 2) P7-Офис 7 и выше либо LibreOffice 7 и выше
Практические занятия	Аудитория	Количество посадочных мест, оборудованных персональными IBM-совместимыми компьютерами Pentium или выше в соответствии с контингентом, рабочее место преподавателя, IBM-совместимый компьютер Pentium или выше, проектор, экран, меловая или маркерная доска	1) Linux Альт Образование 10 и выше; 2) P7-Офис 7 и выше либо LibreOffice 7 и выше
Самостоятельная работа	Помещение для самостоятельной работы	Оснащено компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду университета.	1) Linux Альт Образование 10 и выше; 2) P7-Офис 7 и выше либо LibreOffice 7 и выше

## **8 Адаптация рабочей программы для лиц с ОВЗ**

Адаптированная программа разрабатывается при наличии заявления со стороны обучающегося (родителей, законных представителей) и медицинских показаний (рекомендациями психолого-медико-педагогической комиссии). Для инвалидов адаптированная образовательная программа разрабатывается в соответствии с индивидуальной программой реабилитации.

## ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

<b>№ п/п</b>	<b>Дата</b>	<b>Изменение</b>	<b>Дата и номер протокола заседания УМК</b>	<b>Автор</b>	<b>Начальник ОМОЛА</b>