

**Федеральное государственное автономное образовательное  
учреждение высшего образования  
«Московский физико-технический институт  
(национальный исследовательский университет)»**

**УТВЕРЖДЕНО**

**Директор физтех-школы  
прикладной математики и  
информатики**

**А.М. Райгородский**

	<b>Рабочая программа дисциплины (модуля)</b>
<b>по дисциплине:</b>	Параллельные и распределенные вычисления
<b>по направлению:</b>	Информатика и вычислительная техника
<b>профиль подготовки:</b>	Системное программирование и прикладная математика Физтех-школа Прикладной Математики и Информатики кафедра алгоритмов и технологий программирования
<b>курс:</b>	3
<b>квалификация:</b>	бакалавр

Семестр, формы промежуточной аттестации: 5 (осенний) - Дифференцированный зачет

Аудиторных часов: 60 всего, в том числе:

лекции: 30 час.

семинары: 30 час.

лабораторные занятия: 0 час.

Самостоятельная работа: 75 час.

Всего часов: 135, всего зач. ед.: 3

Количество контрольных работ, заданий: 2

Программу составили:

А.А. Драль, старший преподаватель

О.Н. Ивченко, заведующий кафедрой

А.Н. Сальников, канд. физ.-мат. наук, доцент

Я.А. Леонов, ассистент

Р.Г. Липовский, старший преподаватель

Программа обсуждена на заседании кафедры алгоритмов и технологий программирования 02.04.2024

## Аннотация

На данном курсе студенты познакомятся с:

- такими подходами к разработке параллельных программ как MPI, OpenMP;
- вычислениями на графических процессорах с помощью CUDA.
- устройством распределённых очередей и хранилищ (ZooKeeper, Kafka, Cassandra);
- основными фреймворками обработки больших данных и их внутренним устройством.

Курс ориентирован на практику. Предполагается программирование как на семинарах, так и в рамках домашних заданий.

## 1. Цели и задачи

### Цель дисциплины

Познакомить студентов с основами работы многопроцессорных вычислительных систем и дать практический опыт работы с такими системами. Курс состоит из двух модулей, посвященных соответственно параллельным и распределенным системам. В первом модуле рассматриваются системы в практически «идеальных» условиях, где вычислительные узлы и соединения между ними надежные и быстродействующие. Во втором модуле рассматриваются способы построения надежных систем из ненадежных компонент.

### Задачи дисциплины

Ознакомление с типовыми методами исследования и проектирования параллельных и распределенных вычислений; -приобретение навыков применения современных инструментальных средств и технологий исследования и проектирования параллельных и распределенных вычислений.

## 2. Перечень формируемых компетенций

Освоение дисциплины направлено на формирование следующих компетенций:

Код и наименование компетенции	Индикаторы достижения компетенции
ОПК-2 Способен использовать современные информационные технологии и программные средства при решении задач профессиональной деятельности, соблюдая требования информационной безопасности	ОПК-2.2 Знает и умеет применять численные математические методы и прикладное программное обеспечение для решения научных задач в профессиональной области

## 3. Перечень планируемых результатов обучения по дисциплине (модулю)

В результате освоения дисциплины обучающиеся должны

знать:

- виды и классификацию многопроцессорных вычислительных систем;
- принципы построения распределенных хранилищ данных;
- принципы построения распределенных экосистем (Hadoop / Spark);
- разницу между (одно-) серверными базами данных и распределенными базами данных;
- модель асинхронных вычислений и связь со степенью изоляций транзакций;
- теорему Фишер-Линч-Патерсон (FLP-теорема);
- знать базовые принципы работы Paxos / Raft;
- знать алгоритмы синхронизации времени (NTP, Cristian's Algorithm)
- стандартные задачи распределенных вычислений (Multicasts, Failure Detectors, Membership, Consensus, RSM).

уметь:

- пользоваться библиотекой для параллельных вычислений OpenMP;
- пользоваться библиотекой для параллельных вычислений MPI;
- пользоваться распределенной файловой системой HDFS;
- пользоваться распределенным фреймворком вычислений Hadoop;
- пользоваться распределенным хранилищем данных Hive;
- уметь пользоваться примитивами распределенных вычислений Lamport Timestamps, Vector Clocks
- решать задачу консенсуса в синхронной системе;
- пользоваться алгоритмами Paxos / Raft.

владеть:

- навыками работы с многопроцессорными вычислительными системами (параллельными и распределенными вычислительными системами, в частности)
- кругозором в выборе архитектурного решения поставленной задачи.

#### 4. Содержание дисциплины (модуля), структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий

##### 4.1. Разделы дисциплины (модуля) и трудоемкости по видам учебных занятий

№	Тема (раздел) дисциплины	Трудоемкость по видам учебных занятий, включая самостоятельную работу, час.			
		Лекции	Семинары	Лаборат. работы	Самост. работа
1	Вычисления на GPU. Технология CUDA	10	10		30
2	Параллельные вычисления на MPI и OpenMP	10	10		30
3	Распределённые вычисления на больших объемах данных (HDFS, MapReduce, Hive, Spark)	10	10		15
Итого часов		30	30		75
Подготовка к экзамену		0 час.			
Общая трудоёмкость		135 час., 3 зач.ед.			

##### 4.2. Содержание дисциплины (модуля), структурированное по темам (разделам)

Семестр: 5 (Осенний)

###### 1. Вычисления на GPU. Технология CUDA

Архитектура графических процессоров.

Device (графический процессор) и Host. Обмен данными между ними.

Оптимизация вычислений на графических процессорах.

###### 2. Параллельные вычисления на MPI и OpenMP

Что такое параллельные вычисления?

Устройство и основные структуры в MPI Система очередей SLURM.

Особенности OpenMP

Использование MPI и OpenMP в рамках одной программы.

###### 3. Распределённые вычисления на больших объемах данных (HDFS, MapReduce, Hive, Spark)

Распределённые файловые системы (GFS, HDFS). Её составляющие. Их достоинства, недостатки и сфера применения. Чтение и запись в HDFS. HDFS APIs: WebUI, shell, Java API

Парадигма MapReduce. Основная идея, формальное описание. Обзор реализаций. API для работы с Hadoop (Native Java API vs. Streaming), примеры

Типы Join'ов и их реализации в парадигме MR. Паттерны проектирования MR (pairs, stripes, составные ключи). PageRank в MR. Планировщик задач в YARN.

SQL поверх BigData. Фреймворк Hive

## **5. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю)**

Компьютер и мультимедийное оборудование (проектор, звуковая система).

## **6. Перечень рекомендуемой литературы**

### Основная литература

1. Теория и практика параллельных вычислений, учебное пособие / В. П. Гергель. — Москва, ИНТУИТ, 2016.— URL: <https://ibooks.ru/bookshelf/363224/reading> (дата обращения: 25.11.2020). - Полный текст (Режим доступа : из сети МФТИ / Удаленный доступ)

### Дополнительная литература

## **7. Перечень ресурсов информационно-телекоммуникационной сети "Интернет", необходимых для освоения дисциплины (модуля)**

Не используются

## **8. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень необходимого программного обеспечения и информационных справочных систем (при необходимости)**

Компьютерный класс. Каждый компьютер должен иметь выход в интернет и ПО для подключения к удалённым серверам.

## **9. Методические указания для обучающихся по освоению дисциплины (модуля)**

Успешное освоение курса требует большой самостоятельной работы студента. В программе курса приведено минимально необходимое время для работы студента над темой. Самостоятельная работа включает в себя:

- проработку учебного материала (по конспектам лекций, учебной и научной литературе), подготовку ответов на вопросы, предназначенных для самостоятельного изучения, доказательство отдельных утверждений, свойств;
- подготовку к практическим занятиям, выполнение 6 индивидуальных домашних заданий.

Промежуточный контроль знаний проводится в виде письменных опросов (мини-тестов) по теории.

**ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)**

<b>по направлению:</b>	Информатика и вычислительная техника
<b>профиль подготовки:</b>	Системное программирование и прикладная математика Физтех-школа Прикладной Математики и Информатики кафедра алгоритмов и технологий программирования
<b>курс:</b>	3
<b>квалификация:</b>	бакалавр

Семестр, формы промежуточной аттестации: 5 (осенний) - Дифференцированный зачет

**Разработчики:**

А.А. Драль, старший преподаватель  
О.Н. Ивченко, заведующий кафедрой  
А.Н. Сальников, канд. физ.-мат. наук, доцент  
Я.А. Леонов, ассистент  
Р.Г. Липовский, старший преподаватель

## 1. Компетенции, формируемые в процессе изучения дисциплины

Код и наименование компетенции	Индикаторы достижения компетенции
ОПК-2 Способен использовать современные информационные технологии и программные средства при решении задач профессиональной деятельности, соблюдая требования информационной безопасности	ОПК-2.2 Знает и умеет применять численные математические методы и прикладное программное обеспечение для решения научных задач в профессиональной области

## 2. Показатели оценивания компетенций

В результате изучения дисциплины «Параллельные и распределенные вычисления» обучающийся должен:

### знать:

- виды и классификацию многопроцессорных вычислительных систем;
- принципы построения распределенных хранилищ данных;
- принципы построения распределенных экосистем (Hadoop / Spark);
- разницу между (одно-) серверными базами данных и распределенными базами данных;
- модель асинхронных вычислений и связь со степенью изоляций транзакций;
- теорему Фишер-Линч-Патерсон (FLP-теорема);
- знать базовые принципы работы Paxos / Raft;
- знать алгоритмы синхронизации времени (NTP, Cristian's Algorithm)
- стандартные задачи распределенных вычислений (Multicasts, Failure Detectors, Membership, Consensus, RSM).

### уметь:

- пользоваться библиотекой для параллельных вычислений OpenMP;
- пользоваться библиотекой для параллельных вычислений MPI;
- пользоваться распределенной файловой системой HDFS;
- пользоваться распределенным фреймворком вычислений Hadoop;
- пользоваться распределенным хранилищем данных Hive;
- уметь пользоваться примитивами распределенных вычислений Lamport Timestamps, Vector Clocks
- решать задачу консенсуса в синхронной системе;
- пользоваться алгоритмами Paxos / Raft.

### владеть:

- навыками работы с многопроцессорными вычислительными системами (параллельными и распределенными вычислительными системами, в частности)
- кругозором в выборе архитектурного решения поставленной задачи.

## 3. Перечень типовых (примерных) вопросов, заданий, тем для подготовки к текущему контролю

Примеры заданий на контрольную №1.

1. Строгий системный администратор хочет знать, на какие машины кластера (клиент, DataNode, NameNode, SecondaryNameNode) вам нужен доступ для штатной работы с HDFS. Ответ нужно ему обосновать.
2. Чем определяется число мапперов? А число редьюсеров? Можно ли задать эти значения?

Примеры заданий на контрольную №2.

1. Охарактеризуйте Kafka в терминах CAP, объясните свой выбор.
2. Чем отличается HBase от Cassandra в терминах CAP.

## 4. Перечень типовых (примерных) вопросов и тем для проведения промежуточной аттестации обучающихся

1. Инструменты распределенных вычислений

2. Описание распределенной файловой системы HDFS, отказоустойчивость и масштабируемость.
3. Достоинства и недостатки HDFS.
4. Назначение Secondary Namenode в HDFS.
5. Процесс чтения и записи данных в HDFS.
6. Структура кластера Hadoop. Роли серверов. Как обеспечивается отказоустойчивость.
7. Процесс выполнения задачи на кластере. Distributed cache.
8. Формальная модель MapReduce вычислений.
9. Области применения Writable и WritableComparable.
10. Типы компараторов в MapReduce.
11. Какие этапы работы MapReduce-приложения должен реализовать разработчик.
12. MapReduce: Combiner, Comparator, Partitioner.
13. Типы Join в MapReduce.
14. Достоинства и недостатки Hadoop Streaming.
15. Описание системы Hive для работы с большими данным (где хранится метаданные, где хранятся реальные данные, возможности системы).
16. Теория распределенных вычислений
17. Модель вычислений: процессоры и сеть, аналогия с разделяемой памятью. Синхронные/асинхронные вычисления. 18. Разделяемая память/message passing. Синхронизатор. Тезисно объяснить эквивалентность систем без отказов.
19. Типология отказов процессов, отказы сети. Вложенность классов отказов. Определение casual order (happens before) для asynchronous message-passing систем.
20. Определение консенсуса, три его свойства. Альтернативные формулировки консенсуса: Terminating Reliable Broadcast, Weak Interactive Consistency. Их эквивалентность.
21. Теорема Фишера, Линч и Патерсона (FLP) (без доказательства). Решение задачи о консенсусе в синхронной системе. CAP-теорема (с доказательством). Разница между CAP и FLP теоремами.
22. ACID. Степени изоляции транзакций.

#### Критерии оценивания

- оценка «отлично (10)» выставляется студенту, показавшему всесторонние, систематизированные, глубокие знания учебной программы дисциплины и умение уверенно применять их на практике при решении конкретных задач, свободное и правильное обоснование принятых решений;
- оценка «отлично (9)» выставляется студенту, показавшему всесторонние, систематизированные, глубокие знания учебной программы дисциплины и умение применять их на практике при решении конкретных задач, свободное и правильное обоснование принятых решений;
- оценка «отлично (8)» выставляется студенту, показавшему всесторонние систематизированные, глубокие знания учебной программы дисциплины и умение применять их на практике при решении конкретных задач, и правильное обоснование принятых решений;
- оценка «хорошо (7)» выставляется студенту, если он твердо знает материал, грамотно и по существу излагает его, умеет применять полученные знания на практике, но допускает в ответе или в решении задач некоторые неточности;
- оценка «хорошо (6)» выставляется студенту, если он знает материал, грамотно и по существу излагает его, умеет применять полученные знания на практике, но допускает в ответе или в решении задач некоторые неточности;
- оценка «хорошо (5)» выставляется студенту, если он знает материал, и по существу излагает его, умеет применять полученные знания на практике, но допускает в ответе или в решении задач некоторые неточности;
- оценка «удовлетворительно (4)» выставляется студенту, показавшему фрагментарный, разрозненный характер знаний, недостаточно правильные формулировки базовых понятий, нарушения логической последовательности в изложении программного материала, но при этом он владеет основными разделами учебной программы, необходимыми для дальнейшего обучения и может применять полученные знания по образцу в стандартной ситуации;

- оценка «удовлетворительно (3)» выставляется студенту, показавшему фрагментарный, разрозненный характер знаний, недостаточно правильные формулировки базовых понятий, нарушения логической последовательности в изложении программного материала, но при этом он владеет фрагментарно основными разделами учебной программы, необходимыми для дальнейшего обучения и может применять полученные знания по образцу в стандартной ситуации;
- оценка «неудовлетворительно (2)» выставляется студенту, который не знает большей части основного содержания учебной программы дисциплины, допускает грубые ошибки в формулировках основных понятий дисциплины и не умеет использовать полученные знания при решении типовых практических задач;
- оценка «неудовлетворительно (1)» выставляется студенту, который не знает формулировок основных понятий дисциплины.

## **5. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности**

Во время проведения дифференцированного зачета обучающимся не разрешено пользоваться программой дисциплины.