

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

УТВЕРЖДЕНО
Директор высшей школы
программной инженерии
А.В. Малеев

	Рабочая программа дисциплины (модуля)
по дисциплине:	Алгоритмы во внешней памяти
по направлению:	Программная инженерия
профиль подготовки:	Разработка программно-информационных систем высшая школа программной инженерии высшая школа программной инженерии
курс:	4
квалификация:	бакалавр

Семестр, формы промежуточной аттестации: 7 (осенний) - Экзамен

Аудиторных часов: 60 всего, в том числе:

лекции: 30 час.

семинары: 30 час.

лабораторные занятия: 0 час.

Самостоятельная работа: 18 час.

Подготовка к экзамену: 30 час.

Всего часов: 108, всего зач. ед.: 3

Программу составил: А.А. Жмуров, канд. физ.-мат. наук, доцент

Программа обсуждена на заседании высшей школы программной инженерии 19.03.2025

Аннотация

Курс посвящен изучению алгоритмов и структур данных, оптимизированных для работы с внешней памятью, а также техникам их эффективного применения. Студенты познакомятся с основными моделями вычислений во внешней памяти, методами сортировки, а также структурами данных, такими как B-деревья, хэш-таблицы и Buffered Tree. В ходе курса будут рассмотрены техники оптимизации, включая LSM, Bloom и Xor фильтры, а также методы работы с потоками данных и кэшами. Курс ориентирован на разработку эффективных решений для работы с большими объемами данных и вычислениями в условиях ограниченной памяти.

1. Цели и задачи

Цель дисциплины

Формирование знаний и навыков разработки эффективных алгоритмов и структур данных для работы с внешней памятью, с учётом особенностей памяти и кэширования в современных вычислительных системах.

Задачи дисциплины

- изучить модель вычислений во внешней памяти и методы сортировки;
- освоить структуры данных (B-деревья, хэш-таблицы, Buffered Tree) для работы с внешней памятью;
- применять методы оптимизации, включая LSM, Bloom и Xor фильтры;
- разработать алгоритмы для работы с потоками данных и кэшами (Misra-Gries, Count-Min Sketch);
- изучить методы работы с памятью и кэшами в процессорах и GPU.

2. Перечень формируемых компетенций

Освоение дисциплины направлено на формирование следующих компетенций:

Код и наименование компетенции	Индикаторы достижения компетенции
УК-1 Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	УК-1.2 Находит, критически анализирует и выбирает информацию, необходимую для решения поставленной задачи
	УК-1.3 Рассматривает различные варианты решения задачи, оценивает их преимущества и недостатки
ОПК-1 Способен применять естественнонаучные и общинженерные знания, методы математического анализа и моделирования, теоретического и экспериментального исследования в профессиональной деятельности	ОПК-1.1 Способен анализировать поставленную задачу, намечать пути ее решения
	ОПК-1.3 Способен определять границы применимости полученных результатов
ОПК-6 Способен разрабатывать алгоритмы и программы, пригодные для практического использования, применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов	ОПК-6.3 Знает методы тестирования программного кода на ошибки и способен проводить тестирование на различных уровнях (модульное, интеграционное, системное)
ОПК-7 Способен применять в практической деятельности основные концепции, принципы, теории и факты, связанные с информатикой	ОПК-7.1 Обладает навыками создания и выполнения тестовых сценариев для выявления ошибок в программном обеспечении
	ОПК-7.2 Понимает принципы работы баз данных и умеет проектировать структуру данных для эффективного хранения информации
ОПК-8 Способен осуществлять поиск, хранение, обработку и анализ информации из различных источников и баз данных, представлять ее в требуемом формате с использованием информационных технологий	ОПК-8.1 Понимает принципы, по которым работают базы данных, и умеет создавать структуру данных, оптимизированную для эффективного хранения и обработки информации
	ОПК-8.2 Умеет применять технологии машинного обучения в различных прикладных областях

использованием информационных, компьютерных и сетевых технологий	ОПК-8.3 Умеет оптимизировать и проводить рефакторинг существующего кода для улучшения производительности и поддержки
ПК-3 Способен проектировать, разрабатывать, интегрировать, проверять на работоспособность программное обеспечение	ПК-3.2 Умеет выбирать языки программирования для написания программного кода с учетом технического задания

3. Перечень планируемых результатов обучения по дисциплине (модулю)

В результате освоения дисциплины обучающиеся должны

знать:

- основные модели вычислений во внешней памяти и методы сортировки данных;
- особенности работы с различными структурами данных во внешней памяти (B-деревья, хэш-таблицы, Buffered Tree);
- техники оптимизации работы с внешней памятью, включая LSM, Bloom и Xor фильтры;
- алгоритмы для работы с потоками данных и кэшами (Misra-Gries, Count-Min Sketch);
- принципы работы с памятью и кэшами в современных процессорах и GPU.

уметь:

- применять методы сортировки и алгоритмы для эффективной работы с внешней памятью;
- разрабатывать и анализировать структуры данных, оптимизированные для работы с большими объемами данных;
- использовать техники оптимизации и фильтрации для ускорения обработки данных;
- разрабатывать алгоритмы для обработки данных в потоках и с использованием окон;
- оценивать влияние архитектуры памяти и кэширования на производительность алгоритмов.

владеть:

- навыками разработки и оптимизации алгоритмов для работы с внешней памятью;
- умением проектировать структуры данных для эффективного использования памяти;
- способностью применять современные техники фильтрации и обработки данных для решения практических задач;
- навыками работы с алгоритмами, учитывающими особенности кэширования и работы с потоками данных.

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	Модель вычислений во внешней памяти	3	3		1
2	Сортировка и базовые алгоритмы для работы с внешней памятью	3	3		1
3	Структуры данных для эффективной работы с внешней памятью	3	3		2
4	Хеширование во внешней памяти	3	3		2
5	Техники оптимизации структур данных для внешней памяти	3	3		2
6	Алгоритмы фильтрации запросов и работы с потоками данных	3	3		2
7	Кэширование и модели вычислений с учетом и без учета кэша	3	3		2
8	Оптимизация вычислений в GPU	3	3		2

9	Модели потоковых вычислений и алгоритмы для обработки потоков данных	3	3		2
10	Современные тенденции в разработке алгоритмов для работы с внешней памятью	3	3		2
Итого часов		30	30		18
Подготовка к экзамену		30 час.			
Общая трудоёмкость		108 час., 3 зач.ед.			

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

Семестр: 7 (Осенний)

1. Модель вычислений во внешней памяти

Основные принципы работы с внешней памятью, отличие от классических моделей вычислений, анализ сложности алгоритмов с учётом операций ввода-вывода.

2. Сортировка и базовые алгоритмы для работы с внешней памятью

Внешняя сортировка (двухфазная многоуровневая сортировка, сортировка слиянием), их применение для обработки больших объемов данных.

3. Структуры данных для эффективной работы с внешней памятью

Buffered Tree, B-деревья, B+ ϵ -деревья, их применение в индексировании и обработке больших данных.

4. Хеширование во внешней памяти

Linear hashing, современные подходы к реализации хэш-таблиц для работы с дисковой памятью.

5. Техники оптимизации структур данных для внешней памяти

PGM-index, LSM-деревья и их использование для построения эффективных хранилищ данных.

6. Алгоритмы фильтрации запросов и работы с потоками данных

Bloom и Xor фильтры, Count-Min Sketch, алгоритмы Misra-Gries, BJKST-скетч, их применение в потоковой обработке данных.

7. Кэширование и модели вычислений с учетом и без учета кэша

Cache-oblivious и cache-aware алгоритмы, структура данных COLA, бинарный поиск с учетом и без учета кэша.

8. Оптимизация вычислений в GPU

Архитектура памяти и кэшей в GPU, методы оптимизации доступа к данным для графических процессоров.

9. Модели потоковых вычислений и алгоритмы для обработки потоков данных

Data streaming algorithms, структура данных Q-digest, модель вычислений на окне (Sliding Window).

10. Современные тенденции в разработке алгоритмов для работы с внешней памятью

Новейшие разработки в области эффективной обработки больших данных, оптимизация хранения и поиска информации.

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

Для освоения дисциплины требуется доступ к современным вычислительным системам с возможностью работы с большими объемами данных, включая серверы с внешней памятью и кэшами.

6. Перечень рекомендуемой литературы

Основная литература

Фонд библиотеки МФТИ:

1. Дроздов, С. Н. Структуры и алгоритмы обработки данных: Учебное пособие / Дроздов С.Н. - Таганрог: Южный федеральный университет, 2016. - 228 с.: ISBN 978-5-9275-2242-2. - Текст : электронный. - URL: <https://znanium.com/catalog/product/991928>
2. Григорьев, А. А. Методы и алгоритмы обработки данных : учебное пособие / А.А. Григорьев, Е.А. Исаев. — 2-е изд., перераб. и доп. — Москва : ИНФРА-М, 2024. — 383 с. + Доп. материалы [Электронный ресурс]. — (Высшее образование: Бакалавриат). — DOI 10.12737/1032305. - ISBN 978-5-16-015581-4. - Текст : электронный. - URL: <https://znanium.ru/catalog/product/2084190>

Дополнительная литература

-

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

<http://dm.fizteh.ru/>

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

- языки программирования (C++, Python, Java);
- инструменты для работы с большими данными и потоками;
- средства для моделирования, анализа и визуализации данных;
- системы контроля версий.

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

Для успешного освоения дисциплины рекомендуется:

- изучать теорию, особенно модели вычислений и методы сортировки;
- выполнять практические задания для закрепления знаний о структурах данных и оптимизации;
- осваивать инструменты программирования (C++, Python) для работы с данными;
- дополнительно изучать статьи и курсы для углубления знаний;
- регулярно проверять себя с помощью тестов и контрольных заданий;
- работать в командах для решения практических задач.

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)

по направлению:	Программная инженерия
профиль подготовки:	Разработка программно-информационных систем высшая школа программной инженерии высшая школа программной инженерии
курс:	<u>4</u>
квалификация:	бакалавр

Семестр, формы промежуточной аттестации: 7 (осенний) - Экзамен

Разработчик: А.А. Жмуров, канд. физ.-мат. наук, доцент

1. Компетенции, формируемые в процессе изучения дисциплины

Код и наименование компетенции	Индикаторы достижения компетенции
УК-1 Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	УК-1.2 Находит, критически анализирует и выбирает информацию, необходимую для решения поставленной задачи
	УК-1.3 Рассматривает различные варианты решения задачи, оценивает их преимущества и недостатки
ОПК-1 Способен применять естественнонаучные и общинженерные знания, методы математического анализа и моделирования, теоретического и экспериментального исследования в профессиональной деятельности	ОПК-1.1 Способен анализировать поставленную задачу, намечать пути ее решения
	ОПК-1.3 Способен определять границы применимости полученных результатов
ОПК-6 Способен разрабатывать алгоритмы и программы, пригодные для практического использования, применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов	ОПК-6.3 Знает методы тестирования программного кода на ошибки и способен проводить тестирование на различных уровнях (модульное, интеграционное, системное)
ОПК-7 Способен применять в практической деятельности основные концепции, принципы, теории и факты, связанные с информатикой	ОПК-7.1 Обладает навыками создания и выполнения тестовых сценариев для выявления ошибок в программном обеспечении
	ОПК-7.2 Понимает принципы работы баз данных и умеет проектировать структуру данных для эффективного хранения информации
ОПК-8 Способен осуществлять поиск, хранение, обработку и анализ информации из различных источников и баз данных, представлять ее в требуемом формате с использованием информационных, компьютерных и сетевых технологий	ОПК-8.1 Понимает принципы, по которым работают базы данных, и умеет создавать структуру данных, оптимизированную для эффективного хранения и обработки информации
	ОПК-8.2 Умеет применять технологии машинного обучения в различных прикладных областях
	ОПК-8.3 Умеет оптимизировать и проводить рефакторинг существующего кода для улучшения производительности и поддержки
ПК-3 Способен проектировать, разрабатывать, интегрировать, проверять на работоспособность программное обеспечение	ПК-3.2 Умеет выбирать языки программирования для написания программного кода с учетом технического задания

2. Показатели оценивания компетенций

В результате изучения дисциплины «Алгоритмы во внешней памяти» обучающийся должен:

знать:

- основные модели вычислений во внешней памяти и методы сортировки данных;
- особенности работы с различными структурами данных во внешней памяти (В-деревья, хэш-таблицы, Buffered Tree);
- техники оптимизации работы с внешней памятью, включая LSM, Bloom и Xor фильтры;
- алгоритмы для работы с потоками данных и кэшами (Misra-Gries, Count-Min Sketch);
- принципы работы с памятью и кэшами в современных процессорах и GPU.

уметь:

- применять методы сортировки и алгоритмы для эффективной работы с внешней памятью;
- разрабатывать и анализировать структуры данных, оптимизированные для работы с большими объемами данных;
- использовать техники оптимизации и фильтрации для ускорения обработки данных;
- разрабатывать алгоритмы для обработки данных в потоках и с использованием окон;
- оценивать влияние архитектуры памяти и кэширования на производительность алгоритмов.

владеть:

- навыками разработки и оптимизации алгоритмов для работы с внешней памятью;
- умением проектировать структуры данных для эффективного использования памяти;
- способностью применять современные техники фильтрации и обработки данных для решения практических задач;
- навыками работы с алгоритмами, учитывающими особенности кэширования и работы с потоками данных.

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

Перечень контрольных заданий:

- решение задач на сортировку данных во внешней памяти;
- разработка и анализ структур данных (B-деревья, хэш-таблицы);
- реализация алгоритмов для потоков данных (Misra-Gries, Count-Min Sketch);
- моделирование алгоритмов с учётом кэширования и работы с памятью.

Критерии оценивания:

- корректность решения и реализация алгоритмов;
- оптимизация работы с памятью и эффективное использование ресурсов;
- точность выполнения практических заданий и их соответствие теории;
- умение работать с инструментами и библиотеками.

Методические рекомендации:

- осваивать теоретические материалы перед выполнением практических заданий;
- использовать актуальные библиотеки и инструменты для разработки;
- обсуждать сложные моменты с преподавателями;
- проводить самопроверку после выполнения заданий для выявления ошибок.

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

1. Что такое модель вычислений во внешней памяти и как она отличается от модели в основной памяти?
2. Какие методы сортировки применимы в условиях внешней памяти? Приведите примеры.
3. Объясните принцип работы B-деревьев и их применения в модели внешней памяти.
4. Каковы особенности хэш-таблиц во внешней памяти и что такое техника линейного хеширования?
5. Что такое LSM (Log-Structured Merge) и как эта техника используется для эффективной работы с внешней памятью?
6. Объясните принцип работы Bloom фильтров и их использование в алгоритмах фильтрации запросов.
7. Что такое алгоритм Misra-Gries? Для каких задач он может быть использован?
8. Объясните структуру данных Count-Min Sketch и её применение для подсчёта частоты элементов.
9. В чём разница между cache-oblivious и cache-aware алгоритмами?
10. Как устроены кэши в современных процессорах и как это влияет на разработку алгоритмов?
11. Объясните модель вычислений на окне (Sliding Window) и приведите примеры её применения.
12. Как подсчитать количество различных элементов в потоке данных с использованием BJKST-скетча?

Примеры билетов:

Билет 1.

1. Задача List ranking, техника time forward processing.
2. Хэш-таблицы во внешней памяти, техника linear hashing.

Билет 2.

1. Устройство кэш в современных процессорах, cache-oblivious и cache-aware алгоритмы.
2. Алгоритм Misra-Gries, структура данных Count-Min sketch.

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

Итоговый контроль проводится в формате устного и (или) письменного экзамена.

Время отведенное на экзамен: 4 академических часа.