

**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>	Разработка и анализ алгоритмов
<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>	1
<b>квалификация:</b>	бакалавр

Семестр, формы промежуточной аттестации: 2 (весенний) - Экзамен

Аудиторных часов: 90 всего, в том числе:

лекции: 30 час.

семинары: 0 час.

лабораторные занятия: 60 час.

Самостоятельная работа: 105 час.

Подготовка к экзамену: 30 час.

Всего часов: 225, всего зач. ед.: 5

Количество контрольных работ, заданий: 3

Программу составил: С.Л. Бабичев, канд. физ.-мат. наук, доцент

Программа обсуждена на заседании кафедры алгоритмов и технологий программирования 07.04.2022

## Аннотация

Курс представляет собой первую часть большого курса по разработке и анализу алгоритмов. Она существенно опирается на математический аппарат, полученный студентами на курсе «Математические основания алгоритмов и сложность вычислений».

Семестр посвящен изучению основных структур данных (стек, очередь и пр.) и линейных алгоритмов. Кроме того, изучается мастер-теорема, и все алгоритмы на вводимых структурах данных сопровождаются подробными доказательствами корректности и асимптотики.

## 1. Цели и задачи

### Цель дисциплины

Целями дисциплины являются первичное ознакомление студентов с основными принципами проектирования и анализа алгоритмов и структур данных, обучение навыкам обоснования корректности алгоритмов, их практической реализации, теоретической и экспериментальной оценки их временной сложности.

### Задачи дисциплины

- научить формулировать задачи в терминах изученных теорий, выбирать подходящий алгоритм для поставленной задачи;
- изучить такие разделы, как хеш-таблицы, динамическое программирование и графы;
- изучить такие разделы, как прикладные задачи теории чисел и комбинаторики, поисковые задачи на строках, сжатие текста, вычислительная геометрия и эвристические алгоритмы.
- научить разрабатывать композиции алгоритмов для решения поставленных задач, оценивать сложности алгоритмов, выбирать подходящие структуры данных для поставленных задач, реализовывать алгоритмы на языках программирования C и C++.

## 2. Перечень формируемых компетенций

Освоение дисциплины направлено на формирование следующих компетенций:

Код и наименование компетенции	Индикаторы достижения компетенции
УК-2 Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений	УК-2.1 Формулирует совокупность взаимосвязанных задач в рамках поставленной цели работы, обеспечивающих ее достижение. Определяет ожидаемые результаты решения поставленных задач
	УК-2.2 Проектирует решение конкретной задачи проекта, выбирая оптимальный способ ее решения, исходя из действующих правовых норм и имеющихся ресурсов и ограничений
ОПК-1 Способен применять фундаментальные знания, полученные в области физико-математических и (или) естественных наук, и использовать их в профессиональной деятельности	ОПК-1.1 Способен анализировать поставленную задачу, намечать пути ее решения
ОПК-2 Способен использовать современные информационные технологии и программные средства при решении задач профессиональной деятельности, соблюдая требования информационной безопасности	ОПК-2.1 Способен применять современные вычислительную технику и сервисы сети Интернет в области (сфере) профессиональной деятельности
ОПК-4 Способен осуществлять сбор и обработку научно-технической и (или) технологической информации для решения фундаментальных и прикладных задач	ОПК-4.1 Владеет методами научного поиска и интеллектуального анализа информации при решении задач профессиональной деятельности
	ОПК-4.2 Знает основные источники научно-технической и (или) технологической информации в области профессиональной деятельности
ПК-2 Способен самостоятельно или в качестве члена (руководителя) малого коллектива организовывать и проводить научные исследования и их апробацию	ПК-2.1 Знает принципы построения научной работы, методы сбора и анализа полученного материала, способы аргументации

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

В результате освоения дисциплины обучающиеся должны знать:

- определения асимптотик в O-нотации;
- определения простейших линейных структур данных (стек, очередь, дек, вектор) и сложность обработки запросов в них;
- алгоритм быстрой сортировки;
- алгоритм сортировки слиянием
- определение и практическую необходимость деревьев поиска;
- классификацию и сферы применения хеш-функций;
- организацию и использование хеш-таблиц;
- условия возникновения задачи динамического программирования;
- сведение задачи динамического программирования к функции Беллмана;
- виды задач динамического программирования и методы их решения;
- понятие графов и организацию их хранения;
- алгоритмы обхода графов;
- алгоритмы нахождения специальных элементов в графах;
- алгоритмы нахождения минимальных остовных деревьев;
- алгоритмы нахождения кратчайших путей в графах;
- алгоритмы нахождения потоков в графах;
- алгоритмы нахождения минимальных разрезов и максимальных потоков в графах.
- строго доказывать утверждения о корректности алгоритмов;
- применять необходимую технику для решения алгоритмических задач;
- основы теории чисел в применении к информационным технологиям;
- быстрые алгоритмы работы с длинными числами и матрицами;
- применение алгоритмов теории чисел к задачам криптографии;
- методы комбинаторного поиска;
- Z-функцию и префикс-функцию и их применение;
- структуру данных бор и алгоритм Ахо-Корасик;
- суффиксное дерево и суффиксный автомат;
- алгоритмы сжатия информации: Huffman, LZ77, LZ78, BWT, Arithmetic coding;
- алгоритмы вычислительной геометрии: нахождение выпуклой оболочки, триангуляцию, поиск на плоскости.
- методы приближённого решения NP-сложных задач

уметь:

- оценивать сложность алгоритмов;
- строго доказывать утверждения о корректности алгоритмов;
- применять необходимую технику для решения алгоритмических задач;

владеть:

- разнообразными деревьями поиска и методикой выбора наиболее предпочтительного в каждой конкретной ситуации;
  - методами доказательства корректности утверждений об алгоритмах;
  - методами декомпозиции задач на более простые;
  - методами синтеза решения сложных задач из простых.
- 
- техникой выбора необходимой хеш-функции для данной задачи;
  - техникой сведения уравнения Беллмана к программному коду;
  - техникой реализаций изученных алгоритмов на графах;
  - методами доказательства корректности утверждений об алгоритмах;
  - методами декомпозиции задач на более простые;
  - методами синтеза решения сложных задач из простых.
- 
- техникой выбора необходимых алгоритмов для решения задач теории чисел;
  - техникой быстрого поиска информации в текстовых данных;
  - умением обратимо преобразовывать информацию в более компактное представление;
  - методами аналитической геометрии для решения задач вычислительной геометрии;
  - умением находить приближённые решения сложных задач эвристическими методами.

#### 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	Асимптотики, мастер-теорема.	2		4	9
2	Линейные алгоритмы.	2		4	9
3	Линейные структуры данных.	2		4	9
4	Сортировки и порядковые статистики.	4		8	13
5	Задача поиска.	2		4	9
6	Деревья и кучи.	4		8	12
7	Деревья поиска.	6		12	18
8	Дерево отрезков, дерево Фенвика.	4		8	13
9	Хеш-функции и хеш-таблицы.	4		8	13
Итого часов		30		60	105
Подготовка к экзамену		30 час.			
Общая трудоёмкость		225 час., 5 зач.ед.			

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

###### Семестр: 2 (Весенний)

###### 1. Асимптотики, мастер-теорема.

Обозначения в О-нотации: о-малое и О-большое, омега-малое и Омега-большое, Тета-большое. Независимость определения О-большого и Омега-большого от начального сдвига. Мастер-теорема, пример применения для рекурренты  $T(n) = 3T(n/2) + O(n)$ .

###### 2. Линейные алгоритмы.

Введение в жадные алгоритмы. Критерии применимости жадных алгоритмов. Доказательство корректности жадных алгоритмов. Задачи об аудиториях и о резервных копиях. Понятие об автоматах. Детерминированный конечный автомат.

### 3. Линейные структуры данных.

Структуры данных стек, очередь, вектор, дек. Поиск ближайшего большего справа за  $O(n)$  в массиве. Поиск минимума в стеке и очереди. Метод бухгалтерского учёта для доказательства асимптотики времени обработки запросов в векторе

### 4. Сортировки и порядковые статистики.

Задача сортировки. Определение устойчивой сортировки. Сортировки вставками, Шелла, comb. Сортировка слиянием, подсчёт числа инверсий в перестановке. Сортировка подсчётом, устойчивая сортировка подсчётом, радикас-сортировка. Быстрая сортировка. Варианты Ломута и Хоара. Поиск  $k$ -й порядковой статистики. Дерандомизация: детерминированный алгоритм быстрой сортировки с выбором в качестве пивота медианы массива медиан пятерок.

### 5. Задача поиска.

Обобщённая задача поиска. Линейный поиск. Поиск с сужением зоны. Двоичный и троичный поиск. Подготовка данных для поиска. CRUD-структуры данных. Списки. Списки с пропусками.

### 6. Деревья и кучи.

Определение кучи и запросы, необходимые для обработки. Двоичная куча: операции siftUp и siftDown. Выражение остальных операций через данные. Асимптотика времени работы. Heapsort. Биномиальные деревья и биномиальная куча: скорость работы и преимущества по сравнению с двоичной кучей.

### 7. Деревья поиска.

Определение дерева поиска, обрабатываемые запросы. Теоретическая реализация и анализ времени работы деревьев: splay-деревья, AVL-деревья, декартового дерева, B-деревья как частного случая (a, b)-деревья. Практические применения и преимущества каждого типа деревьев.

### 8. Дерево отрезков, дерево Фенвика.

Обрабатываемые запросы в дереве отрезков. Отложенные операции. Дерево отрезков снизу. Двумерное дерево отрезков. Динамическое и персистентное дерево отрезков. Дерево Фенвика: булевы операции над битами. Многомерное дерево отрезков, запросы к подотрезкам и подпрямоугольникам.

### 9. Хеш-функции и хеш-таблицы.

Хеширование во внешней памяти. Применение хеширования для задач поиска дублирующейся информации. Соокло хеш-таблицы.

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

Учебная аудитория с доской.

## **6. Перечень рекомендуемой литературы**

#### Основная литература

1. Алгоритмы : построение и анализ [Текст] / Т. Кормен, Ч. Лейзерсон, Р. Ривест .— М. : МЦНМО, 2001 .— 960 с.
2. Программирование: теоремы и задачи [Текст], [учеб. пособие] /А. Шень. -М., МЦНМО, 2017
3. Алгоритмы [Текст] : [учеб. пособие для вузов] / С. Дасгупта, Х. Пападимитриу, У. Вазирани ; пер. с англ. А. А. Куликова ; под ред. А. Шеня .— М. : МЦНМО, 2014 .— 320 с.

#### Дополнительная литература

Не предусмотрено

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

1. [http://neerc.ifmo.ru/wiki/index.php?title=Дискретная\\_математика,\\_алгоритмы\\_и\\_структуры\\_данных](http://neerc.ifmo.ru/wiki/index.php?title=Дискретная_математика,_алгоритмы_и_структуры_данных). «Викиконспекты», сайт Санкт-Петербургского Университета ИМТО.
2. <http://e-maxx.ru/>. Maximal algo: личный сайт Максима Иванова, посвящённый алгоритмам и структурам данных.
3. <https://codeforces.com/>. Международная платформа онлайн-контестов по спортивному программированию на базе СГУ и ИМТО.

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

Для успешного освоения дисциплины, студент использует следующие программные средства: компилятор языка C.

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

В преподавании дисциплины сочетаются практические занятия в компьютерном классе и домашние задания, состоящие из теоретических задач и задач на программирование с автоматической проверкой корректности.

**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>	1
<b>квалификация:</b>	бакалавр
Семестр, формы промежуточной аттестации: 2 (весенний) - Экзамен	
<b>Разработчик:</b>	С.Л. Бабичев, канд. физ.-мат. наук, доцент

## 1. Компетенции, формируемые в процессе изучения дисциплины

Код и наименование компетенции	Индикаторы достижения компетенции
УК-2 Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений	УК-2.1 Формулирует совокупность взаимосвязанных задач в рамках поставленной цели работы, обеспечивающих ее достижение. Определяет ожидаемые результаты решения поставленных задач
	УК-2.2 Проектирует решение конкретной задачи проекта, выбирая оптимальный способ ее решения, исходя из действующих правовых норм и имеющихся ресурсов и ограничений
ОПК-1 Способен применять фундаментальные знания, полученные в области физико-математических и (или) естественных наук, и использовать их в профессиональной деятельности	ОПК-1.1 Способен анализировать поставленную задачу, намечать пути ее решения
ОПК-2 Способен использовать современные информационные технологии и программные средства при решении задач профессиональной деятельности, соблюдая требования информационной безопасности	ОПК-2.1 Способен применять современные вычислительную технику и сервисы сети Интернет в области (сфере) профессиональной деятельности
ОПК-4 Способен осуществлять сбор и обработку научно-технической и (или) технологической информации для решения фундаментальных и прикладных задач	ОПК-4.1 Владеет методами научного поиска и интеллектуального анализа информации при решении задач профессиональной деятельности
	ОПК-4.2 Знает основные источники научно-технической и (или) технологической информации в области профессиональной деятельности
ПК-2 Способен самостоятельно или в качестве члена (руководителя) малого коллектива организовывать и проводить научные исследования и их апробацию	ПК-2.1 Знает принципы построения научной работы, методы сбора и анализа полученного материала, способы аргументации

## 2. Показатели оценивания компетенций

В результате изучения дисциплины «Разработка и анализ алгоритмов» обучающийся должен:

**знать:**



- определения асимптотик в О-нотации;
- определения простейших линейных структур данных (стек, очередь, дек, вектор) и сложность обработки запросов в них;
- алгоритм быстрой сортировки;
- алгоритм сортировки слиянием
- определение и практическую необходимость деревьев поиска;
- классификацию и сферы применения хеш-функций;
- организацию и использование хеш-таблиц;
- условия возникновения задачи динамического программирования;
- сведение задачи динамического программирования к функции Беллмана;
- виды задач динамического программирования и методы их решения;
- понятие графов и организацию их хранения;
- алгоритмы обхода графов;
- алгоритмы нахождения специальных элементов в графах;
- алгоритмы нахождения минимальных остовных деревьев;
- алгоритмы нахождения кратчайших путей в графах;
- алгоритмы нахождения потоков в графах;
- алгоритмы нахождения минимальных разрезов и максимальных потоков в графах.
- строго доказывать утверждения о корректности алгоритмов;
- применять необходимую технику для решения алгоритмических задач;
- основы теории чисел в применении к информационным технологиям;
- быстрые алгоритмы работы с длинными числами и матрицами;
- применение алгоритмов теории чисел к задачам криптографии;
- методы комбинаторного поиска;
- Z-функцию и префикс-функцию и их применение;
- структуру данных бор и алгоритм Ахо-Корасик;
- суффиксное дерево и суффиксный автомат;
- алгоритмы сжатия информации: Huffman, LZ77, LZ78, BWT, Arithmetic coding;
- алгоритмы вычислительной геометрии: нахождение выпуклой оболочки, триангуляцию, поиск на плоскости.
- методы приближённого решения NP-сложных задач

#### **уметь:**

- оценивать сложность алгоритмов;
- строго доказывать утверждения о корректности алгоритмов;
- применять необходимую технику для решения алгоритмических задач;

#### **владеть:**

- разнообразными деревьями поиска и методикой выбора наиболее предпочтительного в каждой конкретной ситуации;
- методами доказательства корректности утверждений об алгоритмах;
- методами декомпозиции задач на более простые;
- методами синтеза решения сложных задач из простых.
- техникой выбора необходимой хеш-функции для данной задачи;
- техникой сведения уравнения Беллмана к программному коду;
- техникой реализаций изученных алгоритмов на графах;
- методами доказательства корректности утверждений об алгоритмах;
- методами декомпозиции задач на более простые;
- методами синтеза решения сложных задач из простых.
- техникой выбора необходимых алгоритмов для решения задач теории чисел;
- техникой быстрого поиска информации в текстовых данных;
- умением обратимо преобразовывать информацию в более компактное представление;
- методами аналитической геометрии для решения задач вычислительной геометрии;
- умением находить приближённые решения сложных задач эвристическими методами.

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

для 2-го семестра

1. Опишите алгоритм нахождения числа в упорядоченном массиве за время  $O(\log m)$ , где  $m$  – позиция числа в массиве.
2. Предложите максимально эффективный вариант алгоритма radix-сортировки 32-битных чисел. Обоснуйте своё решение.
3. Предложите реализацию очереди с применением необходимого количества стеков. Определите амортизированную сложность операций enqueue и dequeue.

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

Пример экзаменационных вопросов для 2-го семестра

1. Асимптотики. Операции над асимптотиками. Мастер-теорема.
2. Детерминированный конечный автомат. Определение. Примеры построения.
3. Структура данных стек. Варианты реализации. Асимптотики. Персистентный стек.
4. Структура данных очередь. Варианты реализации. Реализация очереди на стеках.
5. Задача сортировки. Квадратичные сортировки. Инверсии. Сортировки Шелла и comb.
6. Быстрая сортировка. Выбор ведущего элемента. Варианты разбиения на партии Ломута и Хоара. Оценка сложности при рандомизированном выборе ведущего элемента.
7. Алгоритм нахождения  $k$ -й порядковой статистики. Оценка сложности при рандомизированном выборе ведущего элемента.
8. Задача поиска. Линейный поиск и поиск с сужением зоны. Поиск с использованием свойств элемента.
9. Связные списки. CRUD. Операции CRUD в списках. Алгоритмы операций CRUD в структуре данных списки с пропусками.
10. Бинарная куча. Определение и свойства. Операции SiftUp и SiftDown. Асимптотика.
11. Heapsort. Реализация через модифицированные операции SiftUp и SiftDown. Асимптотика. Применимость.
12. Биномиальные деревья. Биномиальная куча. Алгоритмы операций CRUD над биномиальной кучей. Сравнение с бинарной кучей.
13. Деревья поиска. Простое дерево поиска. Асимптотика времени работы рандомизированного времени поиска. Операция вставки в корень. Повороты. Рандомизация деревьев поиска. Удаление узла.
14. Splay-дерево. Операции над splay-деревом. Асимптотика времени работы CRUD-операций. Преимущества и недостатки по сравнению с обычным деревом поиска.
15. Сбалансированные деревья поиска. Варианты балансировки. Оценка высоты дерева в зависимости от организации балансировки.
16. AVL-деревья. Повороты – малый и большой. Балансировка AVL-деревя. Асимптотическая сложность CRUD-операций.
17. Декартовы деревья. Реализация операций split и merge. Алгоритмы CRUD над Декартовыми деревьями.
18. B- и B+-деревья. Организация. Применение. Алгоритмы CRUD над B-деревьями и их сложность.
19. Дерево отрезков. Моноиды. Организация. Прямые операции и их реализация. Отложенные операции. Динамическое дерево отрезков. Персистентное дерево отрезков.
20. Дерево Фенвика. Побитовые операции и их применение для дерева Фенвика. Операции над деревом Фенвика.
21. Обобщённый быстрый поиск. Партиционирование пространства поиска. Хеш-функции. Задача хеширования. Требуемые свойства хеш-функций. Универсальное хеширование.
22. Хеш-таблицы. Организация хеш-таблиц с открытой и закрытой адресацией. Рехеширование. Амортизированная сложность CRUD операций.
23. Задача репликации данных. Дерево репликации. Roll-хеширование.
24. Фильтр Блума. Обоснование эффективности и применение. Реализация. Дедупликация.
25. Докажите, что невозможно создать алгоритм сортировки сравнениями, алгоритмическая сложность которого была бы меньше  $O(N \log N)$ .
26. Сформулируйте и докажите теорему об универсальном семействе хеш-функций.

27. Докажите, что средняя высота случайного дерева поиска есть  $O(N)$ .

Пример экзаменационного билета.

1. Алгоритм нахождения  $k$ -й порядковой статистики. Оценка сложности при рандомизированном выборе ведущего элемента.
2. Организация хеш-таблиц с открытой адресацией. Амортизированная сложность операций поиска, вставки, удаления.

#### Критерии оценивания

отлично

- 10 Полностью и вовремя решены все задачи без ошибок. Продемонстрирован грамотный подход к решению задач, реализованы оптимальные алгоритмы, код оформлен в едином удобочитаемом стиле
- 9 Полностью и вовремя решены все задачи без ошибок. Продемонстрирован грамотный подход к решению задач, реализованы оптимальные алгоритмы
- 8 Полностью и вовремя решены все задачи без ошибок. Продемонстрирован грамотный подход к решению задач

хорошо

- 7 Полностью решены все задачи. Допущены несущественные ошибки.
- 6 Полностью решено большинство задач. В некоторых задачах допущены и не исправлены ошибки, либо некоторые задачи решены частично.
- 5 Полностью решено две трети задач. В некоторых задачах допущены и не исправлены ошибки, либо некоторые задачи решены частично.

удовлетворительно

- 4 Полностью решено более половины задач. В остальных задачах допущены и не исправлены ошибки, либо некоторые задачи решены частично.
- 3 Полностью решено более половины задач.

неудовлетворительно

- 2 Решено менее половины задач.
- 1 Не решено ни одной задачи.

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

При проведении устного экзамена обучающемуся предоставляется 30 минут на подготовку. Опрос обучающегося по билету на устном экзамене не должен превышать двух астрономических часов.

Во время проведения экзамена обучающиеся могут пользоваться программой дисциплины.