ЦЕНТРАЛЬНЫЙ УНИВЕРСИТЕТ

**УТВЕРЖДЕНА**

Решением Ученого совета
АНО ВО «Центральный университет»
«24 июня 2025 г.
Протокол №2

**Рабочая программа дисциплины (модуля)**
**«Алгоритмы и структуры данных. Часть 1. Продвинутый уровень»**

**Направление подготовки:** *02.03.01 Математика и компьютерные науки*

**Направленность (профиль) подготовки:** *Математика и искусственный интеллект*

**Квалификация (степень) выпускника:** *бакалавр*

**Форма обучения:** *очная*

**Срок освоения программы:** *4 года*

**Год набора:** *2025*

ДОКУМЕНТ ПОДПИСАН
ЭЛЕКТРОННОЙ ПОДПИСЬЮ

Сертификат: 0215B0C000D1B3FEB44A37EE631E0C27D4
Владелец: Ивашкевич Евгений Васильевич
Действителен: с 13.01.2026 до 13.04.2027

Электронный документ

**Москва**
**2025**

**Содержание**

1. **Краткая характеристика дисциплины (модуля)** ..... 3
2. **Перечень планируемых результатов обучения** ..... 5
3. **Тематический план** ..... 6
4. **Содержание дисциплины (модуля)** ..... 7
5. **Учебно-методическое обеспечение** ..... 8
6. **Материально-техническое обеспечение** ..... 8
7. **Методические и оценочные материалы** ..... 10

## 1. Краткая характеристика дисциплины (модуля)

Рабочая программа дисциплины (модуля) «Алгоритмы и структуры данных. Часть 1. Продвинутый уровень» составлена в соответствии с федеральным государственным образовательным стандартом высшего образования – бакалавриат по специальности 02.03.01 «Математика и компьютерные науки», профиль «Математика и искусственный интеллект», утвержденный приказом Министерства науки и высшего образования Российской Федерации № 807 от 23.08.2017 года.

Изучение дисциплины (модуля) «Алгоритмы и структуры данных. Часть 1. Продвинутый уровень» формирует у студентов умение разрабатывать и создавать оптимизированные решения, что критически важно для производительности программного обеспечения. Кроме того, эти знания являются основой для дальнейшего изучения более сложных тем в области компьютерных наук и программирования.

**Место дисциплины (модуля) в структуре образовательной программы**

Настоящая дисциплина (модуль) включена в учебный план по программе подготовки бакалавриата по направлению 02.03.01 «Математика и компьютерные науки», профиль «Программа двух дипломов НИУ ВШЭ и ЦУ «Прикладная математика и информатика» и входит в обязательную часть Блока 1, как дисциплина по выбору.

Дисциплина (модуль) изучается на 2 курсе в 3 семестре. Доступна к изучению после успешного освоения любой из дисциплин (модулей): «Язык программирования Java», «Язык программирования C++», «Язык программирования Go», «Язык программирования Python. Базовый», «Язык программирования Python. Основной».

**Цель изучения дисциплины (модуля):** заключается в формировании у студентов навыков проектирования и анализа эффективных алгоритмов и структур данных для решения различных задач программирования.

**Задачи изучения дисциплины (модуля):**

— изучить основные структуры данных, такие как массивы, списки, деревья и хэш-таблицы, их свойства и области применения;

— освоить фундаментальные алгоритмы сортировки, поиска и обработки данных, включая их реализацию на различных языках программирования;

— развить навыки анализа временной и пространственной сложности алгоритмов, а также умения оптимизировать их для повышения эффективности программ.

**В результате освоения дисциплины (модуля) обучающийся должен:**

***знать:***

— модели вычислений и способы оценки вычислительной сложности алгоритмов;

— принципы работы алгоритмов сортировки;

— устройство линейных структур данных;

— свойства хеш-функций и методы разрешения коллизий.

***уметь:***

— выполнять асимптотический анализ;

— выбирать алгоритм сортировки в зависимости от структуры данных и ограничений задачи;

— выбирать подходящую реализацию линейных структур данных в зависимости от задачи;

— проектировать и реализовывать хеш-таблицы;

— основные алгоритмы теории чисел;

— применять теоретико-числовые методы для решения алгоритмических задач.

***владеть:***

— методами формального анализа алгоритмов;

— практикой разработки эффективных алгоритмов обработки массивов;

— методами проектирования эффективных структур данных;
— современными подходами к хранению и поиску данных большого объёма;
— методами построения эффективных алгоритмов для задач теории чисел.

## 2. Перечень планируемых результатов обучения

Компетенции, формируемые в результате освоения дисциплины (модуля) при проведении учебных занятий в форме контактной работы обучающихся с педагогическими работниками Университета и в форме самостоятельной работы обучающихся:

| Компетенция | Содержание компетенции | Индикатор компетенции | Перечень планируемых результатов обучения по дисциплине (модулю) |
|---|---|---|---|
| ОПК-4. | Способен находить, анализировать, реализовывать программно и использовать на практике математические алгоритмы, в том числе с применением современных вычислительных систем | ОПК-4.1. | Знает базовые основы современного математического аппарата, связанного с проектированием, разработкой, реализацией и оценкой качества программных продуктов и программных комплексов в различных областях человеческой деятельности. |
| | | ОПК-4.2. | Умеет использовать этот математический аппарат в профессиональной деятельности. |
| | | ОПК-4.3. | Имеет практический опыт применения современного математического 3 аппарата, связанного с проектированием, разработкой, реализацией и оценкой качества программных продуктов и программных комплексов в различных областях человеческой деятельности. |
| ПК-3 | Способен применять методы математического и алгоритмического моделирования для решения как теоретических, так и практических задач в рамках профессиональной деятельности | ПК-3.1. | Знает основные методы математического и алгоритмического моделирования, а также их применение для решения теоретических и прикладных задач |
| | | ПК-3.2. | Умеет применять методы математического и алгоритмического моделирования для анализа и решения различных задач в области математики и компьютерных наук |
| | | ПК-3.3. | Имеет опыт использования методов математического и алгоритмического моделирования при решении теоретических и прикладных задач в профессиональной деятельности |

## 3. Тематический план

| №п/п | Наименование раздела дисциплины (модуля) | Трудоемкость, академические часы | | | | ТКУ (текущий контроль успеваемости) |
|---|---|---|---|---|---|---|
| | | *Очная форма* | | | | |
| | | Аудиторная работа | | Контроль | Самостоятельная работа | |
| | | Лекции | Семинары | | | |
| 1 | Понятие алгоритма. Оценка трудоемкости алгоритма. Асимптотика. Базовые алгоритмы | 4 | 4 | | 14 | Домашнее задание |
| 2 | Теория чисел. Часть 1. | 2 | 2 | | 14 | Домашнее задание |
| 3 | Алгоритмы сортировки и поиска | 6 | 2 | 4 | 18 | Домашнее задание<br>Контрольная работа<br>Коллоквиум<br>Лабораторная работа |
| 4 | Структуры данных | 2 | 2 | | 30 | Домашнее задание |
| 5 | Теория чисел. Часть 2. | 4 | 4 | | 22 | Домашнее задание |
| 6 | Хеширование | 6 | 4 | 2 | 14 | Домашнее задание<br>Контрольная работа |
| 7 | Динамическое программирование | 4 | 2 | 2 | 14 | Домашнее задание<br>Контрольная работа<br>Лабораторная работа |
| | ***Экзамен*** | | | **4** | | |
| | ***Итого*** | ***30*** | ***22*** | ***12*** | ***126*** | |
| ***Объем дисциплины (модуля) (в ак. ч.)*** | | ***190*** | | | | |
| ***Объем дисциплины (модуля) (в зач. ед.)*** | | ***5*** | | | | |

## 4. Содержание дисциплины (модуля)

| №п/п | Наименование раздела дисциплины (модуля) | Содержание дисциплины (модуля) по темам |
|---|---|---|
| 1 | Понятие алгоритма. Оценка трудоемкости алгоритма. Асимптотика. Базовые алгоритмы | Понятие алгоритма. Способы измерения времени выполнения алгоритмов. Асимптотические нотации. Мастер-теорема. Реккурентные соотношения. Линейные алгоритмы. Рекурсивные алгоритмы (генерация комбинаторных объектов) |
| 2 | Теория чисел. Часть 1. | Теория чисел. Часть 1. |
| 3 | Алгоритмы сортировки и поиска | Бинарный поиск. Сортировка слиянием. Быстрая сортировка. k-ая порядковая статистика. Медиана медиан. Куча. Сортировка кучей |
| 4 | Структуры данных | Односвязный список. Стек. Очередь. Амортизационный анализ. Динамический массив. Очередь на двух стеках |
| 5 | Теория чисел. Часть 2. | Теория чисел. Часть 2. Быстрое преобразование Фурье |
| 6 | Хеширование | Хеширование |
| 7 | Динамическое программирование | Динамическое программирование |

## 5. Учебно-методическое обеспечение

Университет располагает полным набором лицензионного и свободно распространяемого программного обеспечения, включая продукты отечественного производства.

Каждый студент в течение всего периода обучения получает индивидуальный неограниченный доступ к электронно-библиотечной системе и электронной информационно-образовательной среде университета. Эти системы предоставляют возможность доступа к ресурсам из любой точки, где есть подключение к сети Интернет, как на территории университета, так и за его пределами.

Студентам обеспечен удаленный доступ к современным профессиональным базам данных и информационным справочным системам.

***Основная литература:***

1. Протодьяконов, А. В. Алгоритмы Data Science и их практическая реализация на Python : учебное пособие / А. В. Протодьяконов, П. А. Пылов, В. Е. Садовников. - Москва ; Вологда : Инфра-Инженерия, 2022. - 392 с. - ISBN 978-5-9729-1006-9. - Текст : электронный. - URL: https://znanium.com/catalog/product/1902689.

2. Бхаргава, А. Грокаем алгоритмы. Иллюстрированное пособие для программистов и любопытствующих : пособие / А. Бхаргава. - Санкт-Петербург : Питер, 2021. - 288 с. - (Серия «Библиотека программиста»). - ISBN 978-5-4461-0923-4. - Текст : электронный. - URL: https://znanium.com/catalog/product/1739631.

3. Черпаков, И. В. Алгоритмизация и программирование на Python : учебник для вузов / И. В. Черпаков. — Москва : Издательство Юрайт, 2026. — 159 с. — (Высшее образование). — ISBN 978-5-534-21910-4. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/582412.

4. 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:***

1. Палий, И. А. Линейное программирование : учебник для вузов / И. А. Палий. — 2-е изд., испр. и доп. — Москва : Издательство Юрайт, 2025. — 175 с. — (Высшее образование). — ISBN 978-5-534-04716-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/563472.

2. Методы оптимизации: теория и алгоритмы : учебник для вузов / А. А. Черняк, Ж. А. Черняк, Ю. М. Метельский, С. А. Богданович. — 2-е изд., испр. и доп. — Москва : Издательство Юрайт, 2025. — 354 с. — (Высшее образование). — ISBN 978-5-534-04103-3. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/559380.

3. Сухарев, А. Г. Численные методы оптимизации : учебник и практикум для вузов / А. Г. Сухарев, А. В. Тимохов, В. В. Федоров. — 3-е изд., испр. и доп. — Москва : Издательство Юрайт, 2025. — 367 с. — (Высшее образование). — ISBN 978-5-534-17381-9. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/562179.

## 6. Материально-техническое обеспечение

Университет располагает материально-технической базой, соответствующей действующим противопожарным правилам и нормам и обеспечивающей проведение всех

видов дисциплинарной и междисциплинарной подготовки, практической и научно-исследовательской работ обучающихся, предусмотренных учебным планом.

Помещения, которые представляют собой учебные аудитории для проведения занятий лекционного типа, занятий семинарского (практического) типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, а также помещения для самостоятельной работы и помещения для хранения и профилактического обслуживания учебного оборудования. Помещения укомплектованы специализированной мебелью и техническими средствами обучения, служащими для представления учебной информации большой аудитории.

Изучение дисциплины (модуля) обеспечивается в учебных аудиториях, оснащенных:

— столами и стульями;
— компьютерной техникой;
— механическими калькуляторами;
— специализированным оборудованием, включая демонстрационное оборудование.

Помещения для самостоятельной работы обучающихся, в том числе приспособленные для использования инвалидами и лицами с ограниченными возможностями здоровья, оснащены компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду Университета.

Обучающимся предоставляется доступ (в том числе удаленный) к ресурсам информационно-телекоммуникационной сети «Интернет», электронным ресурсам (в том числе электронным библиотечным системам, современным профессиональным базам данных и информационным справочным системам):

| № | Наименование портала (издания, курса, документа) | Ссылка |
|---|---|---|
| 1. | Научная электронная библиотека elibrary.ru библиотека | https://elibrary.ru/defaultx.asp |
| 2. | База данных для IT-специалистов | https://habr.com |
| 3. | База данных ScienceDirect | https://www.sciencedirect.com |
| 4. | Официальный сайт Министерства науки и высшего образования Российской Федерации | https://minobrnauki.gov.ru/ |
| 5. | Федеральный портал «Российское образование" | https://www.edu.ru/ |
| 6. | Информационная система "Единое окно доступа к образовательным ресурсам" | http://window.edu.ru/ |
| 7. | Единая коллекция цифровых образовательных ресурсов | http://school-collection.edu.ru/ |
| 8. | Федеральный центр информационно - образовательных ресурсов | http://fcior.edu.ru/ |

Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), в том числе комплект лицензионного программного обеспечения, современные профессиональные базы данных и информационные справочные системы:

| Наименование ПО | Производство | Лицензионное / свободно распространяемое |
|---|---|---|
| **Операционные системы:** | | |
| Microsoft Imagine (Windows Client, Server) | зарубежное | лицензионное |
| **Браузеры:** | | |
| Яндекс.Браузер | отечественное | свободно распространяемое |
| Google Chrome | зарубежное | свободно распространяемое |
| **Офисные приложения:** | | |

| | | |
|---|---|---|
| Microsoft Imagine (Visio, OneNote) | зарубежное | лицензионное |
| TeXstudio | зарубежное | свободно распространяемое |
| Adobe Acrobat Reader | зарубежное | свободно распространяемое |
| **Программное обеспечение для планирования и учета времени:** | | |
| Toggle app | зарубежное | свободно распространяемое |
| **Системы управления проектами:** | | |
| Microsoft Imagine (Project) | зарубежное | лицензионное |
| **Системы управления базами данных:** | | |
| Microsoft Imagine (SQL Server) | зарубежное | лицензионное |
| **Системы резервного копирования (backup):** | | |
| Acronis Backup Advanced for HyperV | зарубежное | лицензионное |
| **Справочно-правовые системы:** | | |
| КонсультантПлюс: справочно-правовая система | отечественное | лицензионное |
| **Средства антивирусной защиты:** | | |
| Kaspersky Endpoint Security для бизнеса Стандартный Russian Edition | отечественное | лицензионное |
| **Среды разработки:** | | |
| Visual Studio Code | зарубежное | свободно распространяемое |
| Bash (Unix shell) | зарубежное | свободно распространяемое |
| Anaconda | зарубежное | свободно распространяемое |
| Robotic Operating System | зарубежное | свободно распространяемое |
| CopelliaSim | зарубежное | свободно распространяемое |
| Google Colaboratory | зарубежное | свободно распространяемое |
| **Пакеты программных средств и библиотек:** | | |
| AutoPsy | зарубежное | свободно распространяемое |
| Interactive Disassembler (IDA) | зарубежное | свободно распространяемое |
| **Системы управления библиографической информацией:** | | |
| Zotero | зарубежное | свободно распространяемое |
| **Сервисы и службы:** | | |
| Bind | зарубежное | свободно распространяемое |
| Docker | зарубежное | свободно распространяемое |

## 7. Методические и оценочные материалы

### Методические указания для обучающихся по освоению дисциплины (модуля)

В процессе изучения дисциплины (модуля) «Алгоритмы и структуры данных. Часть 1. Продвинутый уровень» в рамках текущего контроля успеваемости используются такие виды учебной работы, как лекция, семинары, домашние задания, коллоквиум, лабораторные работы, контрольная работа, а также различные виды самостоятельной работы обучающихся по заданию преподавателя, направленные на развитие навыков профессиональной лексики, закрепление практических профессиональных компетенций, поощрение инициатив.

*Лекция* – систематическое, последовательное, монологическое изложение преподавателем учебного материала, как правило, теоретического характера.

В процессе лекций рекомендуется вести конспект лекций: кратко и схематично фиксировать основные идеи, выводы и обобщения лекции; выделять важные мысли, ключевые слова и термины. Необходимо отметить вопросы или материалы, которые вызывают затруднения, и попытаться найти ответы в рекомендованной литературе. Если разобраться в материале не удается, следует сформулировать вопрос и задать его преподавателю на консультации или во время семинарского (практического) занятия.

*Семинар* – это форма учебной деятельности, проводимая в учебном заведении под руководством преподавателя, где студенты активно участвуют в обсуждениях, практических заданиях и других формах взаимодействия.

Для успешной подготовки к семинару рекомендуется заранее ознакомиться с темой занятия и основными материалами, чтобы иметь возможность активно участвовать в обсуждении. Также полезно подготовить вопросы и идеи для обсуждения, что поможет глубже понять материал и продемонстрировать заинтересованность.

*Коллоквиум* – устные ответы на вопросы, список которых известен студенту заранее,

В процессе подготовки к коллоквиуму необходимо проанализировать учебные материалы, ознакомившись с лекциями, учебниками и дополнительными источниками, акцентируя внимание на ключевых темах. Рекомендуется создать структурированные конспекты, выделяя основные идеи, термины и формулы.

*Контрольная работа* – письменная работа с набором задач, которые нужно решить за ограниченное время.

Цель контрольной работы - получить специальные знания по одной или нескольким темам дисциплины (модуля) и продемонстрировать навыки их практического применения.

*Лабораторная работа* – задания, направленные на практическое закрепление теоретических знаний через выполнение экспериментальных или проектных заданий. В ходе лабораторной работы студенты проводят анализ, разработку или тестирование программного кода, изученного в рамках курса.

Перед выполнением лабораторной работы необходимо внимательно ознакомиться с теоретическим материалом и методическими указаниями. Рекомендуется изучить используемые инструменты и технологии для уверенного их применения на практике. При возникновении вопросов следует обращаться к преподавателю для получения разъяснений.

*Домашнее задание* – набор задач по темам недели.

При работе над домашними заданиями важно внимательно ознакомиться с требованиями и сроками выполнения. Рекомендуется разбивать задания на этапы, чтобы избежать перегрузки и лучше усвоить материал. Использовать различные источники информации, включая учебники и онлайн-ресурсы, для более глубокого понимания темы.

*Самостоятельная работа* – работа студентов, направленная на углубленное изучение отдельных тем и вопросов учебной дисциплины (модуля).

В процессе самостоятельной работы студенты взаимодействуют с рекомендованными материалами при минимальном участии преподавателя. Задачи студента включают работу с конспектами лекций (обработка текста), повторное изучение учебных материалов планов и тезисов ответов, изучение дополнительных тем, выполнение учебно-исследовательских заданий и другое.

## Система оценивания результатов обучения по дисциплине (модулю)

**Критерии получения уровня и оценивания сформированности компетенций по дисциплине (модулю) «Алгоритмы и структуры данных. Часть 1. Продвинутый уровень»**

Оценивание уровня учебных достижений, обучающихся по дисциплине (модулю), осуществляется в виде текущего контроля успеваемости и промежуточной аттестации.

**Промежуточная аттестация** по дисциплине (модулю) осуществляется в форме ***экзамена***, при этом проводится оценка компетенций, сформированных по дисциплине.

Для оценивания текущего контроля успеваемости и промежуточной аттестации

используется десятибалльная шкала оценивания, которая соотносится с традиционной пятибалльной шкалой следующим образом, если иное не указано в силлабусе дисциплины:

| **Десятибалльная оценка** | **Пятибалльная оценка** | **Общая характеристика результата обучения по дисциплине (модулю)** |
|---|---|---|
| 10 | Отлично | Студент полностью владеет знаниями, изложенными в рабочей программе, и глубоко осмысляет дисциплину. Он самостоятельно и логически последовательно отвечает на все вопросы, акцентируя внимание на наиболее важном. Умеет анализировать, сравнивать, классифицировать, обобщать, конкретизировать и систематизировать изученный материал, выделяя ключевые моменты и устанавливая причинно-следственные связи. Четко формулирует ответы, уверенно интерпретирует результаты анализов и других исследований, а также решает сложные задачи. Студент хорошо знаком с методами исследования, необходимыми для практической деятельности, и умеет связывать теоретические аспекты дисциплины (модуля) с практическими задачами. |
| 9 | Отлично | |
| 8 | Отлично | |
| 7 | Хорошо | Студент обладает знаниями предмета почти в полном объеме рабочей программы и самостоятельно, логически последовательно и всесторонне отвечает на все вопросы, акцентируя внимание на наиболее значимых моментах. Он умеет анализировать, сравнивать, классифицировать, обобщать, конкретизировать и систематизировать изученный материал, выделяя его ключевые аспекты и устанавливая причинно-следственные связи. Формулирует свои ответы, уверенно интерпретирует результаты анализов и других исследований, а также решает сложные ситуационные задачи. Студент хорошо знаком с методами исследования, необходимыми для практической деятельности, и умеет связывать теоретические аспекты предмета с практическими задачами. |
| 6 | Хорошо | |
| 5 | Удовлетворительно | Студент обладает базовыми знаниями по дисциплине (модулю), но испытывает трудности при самостоятельных ответах и использует неточные формулировки. В ходе ответов он допускает ошибки, касающиеся сути вопросов. Студент способен решать только самые простые задачи и владеет лишь минимальным набором методов исследования. |
| 4 | Удовлетворительно | |
| 3 | Не сдан | Студент не овладел обязательным минимумом знаний по предмету и не может ответить на вопросы, даже если преподаватель задает дополнительные наводящие вопросы. |
| 2 | Не сдан | |
| 1 | Не сдан | |

Дисциплина (модуль) «Алгоритмы и структуры данных. Часть 1. Продвинутый уровень» оценивается следующим образом:

| Активность | Вес | Количество | Описание |
|---|---|---|---|
| Домашние задания | 15% | 12 | Набор задач по темам недели |
| Лабораторная работа | 10% | 2 | Практическая работа по темам дисциплины (модуля) |
| Контрольная работа | 30% | 3 | Письменная работа с набором задач, которые нужно решить за ограниченное время |
| Коллоквиум | 15% | 1 | Устные ответы на вопросы, список которых известен студенту заранее |
| Экзамен | 30% | 1 | Письменная или устная работа над заданием, направленным на проверку полученных знаний и навыков по дисциплине (модулю) |

**Формула расчёта итоговой оценки по дисциплине (модулю) «Алгоритмы и структуры данных. Часть 1. Продвинутый уровень»:** «0,15 х среднее за домашние задания + 0,1 х среднее за лабораторные работы +0,2 х среднее за контрольные работы + 0,15 х коллоквиум + 0,3 х экзамен».

**Текущий контроль успеваемости обучающихся по дисциплине (модулю)**
**Примерные вопросы к коллоквиуму**

**Классификация структур данных**

1. **Структуры данных** - определение и классификация.
2. **Линейные структуры данных** - описание и примеры.
3. **Нелинейные структуры данных** - описание и примеры.
4. **Статические и динамические структуры данных** - различия и примеры.
5. **Абстрактные типы данных (АТД)** - определение и примеры.
6. **Комплексные структуры данных** - определение и примеры.

**Стеки**

7. **Стек** - определение и основные операции (push, pop).
8. **Применение стека** - примеры использования (обратная польская запись, отмена действий).
9. **Алгоритм проверки сбалансированности скобок** - описание и реализация.
10. **Реализация стека с помощью массива** - описание и анализ.
11. **Реализация стека с помощью связного списка** - описание и анализ.

**Очереди**

12. **Очередь** - определение и основные операции (enqueue, dequeue).
13. **Кольцевая очередь** - определение и реализация.
14. **Приоритетная очередь** - определение и применение.
15. **Очередь с двумя концами (дека)** - определение и операции.

**Множества и словари**

16. **Множество** - определение и основные операции (объединение, пересечение).
17. **Словарь (ассоциативный массив)** - определение и применение.
18. **Хеш-таблица** - определение, функции хеширования и коллизии.
19. **Алгоритм поиска в множестве** - различные подходы и их сложность.
20. **Динамические множества** - реализация и применение.

**Односвязные линейные списки**

21. **Односвязный список** - определение и основные операции (добавление, удаление).
22. **Реверсирование односвязного списка** - алгоритм и сложность.
23. **Поиск элемента в односвязном списке** - алгоритм и сложность.
24. **Слияние двух отсортированных списков** - алгоритм и реализация.
25. **Удаление дубликатов из односвязного списка** - алгоритм и реализация.

**Циклические списки**

26. **Циклический односвязный список** - определение и операции.

27. **Поиск элемента в циклическом списке** - алгоритм и сложность.
28. **Слияние циклических списков** - алгоритм и реализация.

**Двусвязные линейные списки**

29. **Двусвязный список** - определение и основные операции.
30. **Преимущества и недостатки двусвязного списка по сравнению с односвязным** - анализ.
31. **Реверсирование двусвязного списка** - алгоритм и сложность.
32. **Удаление элемента из двусвязного списка** - алгоритм и реализация.

**Общие темы**

33. **Сравнение структур данных** - когда использовать каждую из них.
34. **Сложность операций** - временная и пространственная сложность для различных структур.
35. **Реализация структуры данных в различных языках программирования** - примеры.
36. **Применение структур данных в алгоритмах** - примеры использования.
37. **Алгоритмы на графах** - связь с линейными списками.
38. **Структуры данных для реализации очередей и стеков** - различные подходы.
39. **Динамическое выделение памяти** - работа с указателями.
40. **Обзор популярных библиотек для работы со структурами данных** - примеры и применение.

## Примерные задания для контрольных работ

### Контрольная работа №1

**Задание 1.** Определите, что такое алгоритм. Приведите примеры различных типов алгоритмов и их применения в реальной жизни.

**Задание 2.** Опишите основные способы измерения времени выполнения алгоритмов. Сравните их и объясните, в каких случаях каждый из методов наиболее уместен.

**Задание 3.** Объясните, что такое асимптотическая нотация. Приведите примеры использования нотаций $O(n)$, $\Theta(n)$ и $\Omega(n)$ и объясните, что они означают.

**Задание 4.** Примените мастер-теорему для анализа временной сложности следующего рекуррентного соотношения: $T(n) = 2T(n/2) + n$. Укажите, какой случай мастер-теоремы вы использовали и приведите подробное решение.

**Задание 5.** Решите следующее рекуррентное соотношение с помощью метода подстановки: $T(n) = T(n - 1) + n$. Найдите асимптотику $T(n)$.

**Задание 6.** Напишите алгоритм для линейного поиска элемента в массиве и проанализируйте его временную сложность. Объясните, как вы пришли к этому результату.

**Задание 7.** Опишите алгоритм, который находит сумму всех элементов в односвязном списке. Какова временная сложность этого алгоритма?

**Задание 8.** Приведите пример рекуррентного соотношения, которое можно решить с помощью мастер-теоремы. Проанализируйте его временную сложность и объясните, какой случай теоремы был использован.

**Задание 9.** Объясните, как можно использовать линейные алгоритмы для решения задачи о нахождении максимального и минимального элементов в массиве. Напишите алгоритм и проанализируйте его сложность.

**Задание 10.** Приведите пример задачи, которая может быть решена с помощью рекурсии, и напишите соответствующий рекурсивный алгоритм. Проанализируйте временную сложность вашего решения.

### Контрольная работа №2

**Задание 1.** Опишите алгоритм Евклида для нахождения наибольшего общего делителя (НОД) двух чисел. Приведите пример работы алгоритма на конкретных числах.

**Задание 2.** Реализуйте алгоритм Решета Эратосфена для нахождения всех простых чисел до N. Объясните, как работает этот алгоритм, и приведите пример его использования.

**Задание 3.** Объясните, что такое факторизация чисел. Напишите алгоритм для нахождения простых множителей заданного числа и проанализируйте его временную сложность.

**Задание 4.** Реализуйте рекурсивный алгоритм для бинарного поиска в отсортированном массиве. Приведите пример массива и элемента для поиска, а также объясните, как работает ваш алгоритм.

**Задание 5.** Напишите алгоритм для целочисленного бинарного поиска по ответу. Приведите пример задачи, где этот метод может быть применен, и объясните шаги алгоритма.

**Задание 6.** Приведите примеры базовых сортировок (пузырьковая, вставками, выбором) и проанализируйте их временную сложность в лучшем, среднем и худшем случаях.

**Задание 7.** Опишите алгоритм сортировки слиянием. Реализуйте его и проанализируйте временную и пространственную сложность.

**Задание 8.** Реализуйте алгоритм быстрой сортировки. Объясните, как выбирается опорный элемент и как это влияет на эффективность алгоритма.

**Задание 9.** Объясните метод двух указателей и приведите пример его использования для решения задачи, например, для нахождения пары чисел в массиве, сумма которых равна заданному значению.

**Задание 10.** Опишите алгоритм "скользящее окно" и приведите пример задачи, где он может быть применен, например, для нахождения максимальной суммы подмассива фиксированной длины. Реализуйте алгоритм и проанализируйте его сложность.

## Примерные домашние задания

### Домашнее задание 1.

1. Определите, что такое алгоритм, и приведите три примера различных алгоритмов из повседневной жизни.
2. Назовите и объясните три основных свойства алгоритмов.
3. Сравните детерминированные и недетерминированные алгоритмы. Приведите примеры каждого типа.
4. Опишите, как можно измерить время выполнения алгоритма. Какие инструменты могут быть использованы для этой цели?
5. Объясните, что такое временная сложность алгоритма и как она может повлиять на выбор алгоритма для решения задачи.

### Домашнее задание 2.

1. Что такое асимптотические нотации? Приведите примеры "О большого", "Ω" и "Θ".
2. Объясните, как интерпретировать нотацию "О большого" на примере алгоритма сортировки.
3. Что такое мастер-теорема и в каких случаях она используется? Приведите пример.
4. Решите следующее рекуррентное соотношение: $T(n)=2T(n/2)+n$ $T(n)=2T(n/2)+n$. Используйте мастер-теорему для нахождения временной сложности.
5. Приведите пример рекурсивного алгоритма и объясните его временную сложность.

### Домашнее задание 3.

1. Опишите алгоритм сортировки слиянием. Какова его временная сложность в лучшем и худшем случае?

2. Объясните принцип работы быстрой сортировки. Как она отличается от сортировки слиянием?

3. Как работает бинарный поиск? В каких условиях он применяется?

4. Реализуйте алгоритм бинарного поиска на языке программирования по вашему выбору и протестируйте его на различных входных данных.

5. Объясните, как осуществляется целочисленный и вещественный бинарный поиск по ответу. Приведите примеры.

**Примерные задания для лабораторных работ**

**Лабораторная работа по теме «Теория чисел. Быстрое преобразование Фурье»**

1. Реализуйте умножение двух больших чисел (заданных как строки) с помощью быстрого преобразования Фурье (БПФ). Проверьте корректность, сравнив с обычным умножением.

2. Напишите программу для умножения двух многочленов через БПФ. На вход — коэффициенты, на выход — коэффициенты произведения. Протестируйте на примерах: $(1 + 2x + x^2) \times (x + 3x^2)$.

3. Реализуйте прямое и обратное БПФ для массива комплексных чисел. Убедитесь, что после обратного преобразования данные восстанавливаются с точностью 1e-10.

4. Сравните время умножения чисел из 1000 и 10 000 цифр с помощью:
   - Наивного алгоритма
   - Алгоритма Карацубы
   - Реализации через БПФ
5. Постройте таблицу и сделайте вывод.
6. Напишите отчёт (1–2 стр.), где объясните:
   - Как БПФ ускоряет умножение
   - Где оно применяется (криптография, сигналы, компиляторы)
   - Почему БПФ не используется везде (точность, сложность, накладные расходы)

**Лабораторная работа по теме: «Динамическое программирование»**

1. Реализуйте алгоритм поиска наибольшей возрастающей подпоследовательности (LIS) в массиве. Выведите длину и саму подпоследовательность. Протестируйте на: [10][9][2][5][3][7][101][18].

2. Напишите программу для поиска наибольшей общей подпоследовательности (LCS) двух строк. Восстановите саму подпоследовательность. Протестируйте на: "ABAZDC" и "BACBAD".

3. Решите задачу о рюкзаке (0-1): даны веса, стоимости и грузоподъёмность. Найдите максимальную стоимость и набор предметов. Оцените сложность по времени и памяти.

4. Реализуйте подсчёт числа способов разменять сумму S монетами номиналов [1][2][5]. Используйте динамическое программирование. Сравните по скорости с рекурсией.

5. Напишите отчёт (1–2 стр.), где:
   - Объясните принцип ДП (оптимальная подструктура, перекрывающиеся подзадачи)
   - Приведите примеры применения (биоинформатика, маршрутизация, финансы)
   - Обоснуйте, почему ДП эффективнее полного перебора

**Задания для промежуточной аттестации по дисциплине (модулю)**

| № п/п | Задание | Ответ | Компетенция |
|---|---|---|---|
| 1. | Какая асимптотическая оценка получается при решении рекуррентности T(n) = 2T(n/2) + n по мастер-теореме? | O(n log n) | ОПК-4 |
| 2. | Как называется метод анализа стоимости операций, усредняющий затраты на последовательность действий? | амортизационный анализ | ПК-3 |
| 3. | Какую временную сложность имеет детерминированный алгоритм поиска k-ой порядковой статистики (медиана медиан)? | O(n) | ОПК-4 |
| 4. | Какова асимптотическая сложность умножения многочленов с использованием быстрого преобразования Фурье? | O(n log n) | ОПК-4 |
| 5. | Какова временная сложность решения задачи о наибольшей общей подпоследовательности методом динамического программирования? | O(n*m) | ПК-3 |
| 6. | Чему равна нижняя оценка сложности сортировки сравнением для n элементов? | Omega(n log n) | ОПК-4 |
| 7. | Какой алгоритм позволяет найти обратный элемент по модулю простого числа за логарифмическое время? | расширенный алгоритм Евклида / бинарное возведение | ОПК-4 |
| 8. | Что используется в амортизационном анализе для учета «запаса» стоимости операций? | потенциал / потенциальная функция | ПК-3 |
| 9. | Какое преобразование используется для быстрого умножения больших чисел или многочленов? | быстрое преобразование Фурье / БПФ | ОПК-4 |
| 10. | Как называется свойство хеш-функции, при котором любой ключ равновероятно попадает в любую ячейку? | универсальное хеширование | ОПК-4 |
| 11. | Какова глубина рекурсии при сортировке слиянием в худшем случае? | O(log n) | ОПК-4 |
| 12. | Как называется подход, позволяющий уменьшить пространственную сложность ДП с O(n^2) до O(n)? | оптимизация памяти / хранение только предыдущей строки | ПК-3 |