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 2. Продвинутый уровень»

Направление подготовки: *02.03.01 Математика и компьютерные науки*

Направленность (профиль) подготовки: *Разработка*

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** 4
3. **Тематический план** 5
4. **Содержание дисциплины (модуля)** 6
5. **Учебно-методическое обеспечение** 7
6. **Материально-техническое обеспечение** 7
7. **Методические и оценочные материалы** 9

1. Краткая характеристика дисциплины (модуля)

Рабочая программа дисциплины (модуля) «Алгоритмы и структуры данных. Часть 2. Продвинутый уровень» составлена в соответствии с федеральным государственным образовательным стандартом высшего образования – бакалавриат по специальности 02.03.01 «Математика и компьютерные науки», профиль «Разработка», утвержденный приказом Министерства науки и высшего образования Российской Федерации № 807 от 23.08.2017 года.

Изучение дисциплины (модуля) «Алгоритмы и структуры данных. Часть 2. 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 «Разработка» и входит в обязательную часть Блока 1, как дисциплина по выбору.

Дисциплина (модуль) изучается на 2 курсе в 4 семестре. Доступна к изучению после успешного освоения дисциплины (модуля): «Алгоритмы и структуры данных. Часть 1. Продвинутый уровень».

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:

— принципы работы и области применения продвинутых структур данных (дерево Фенвика, дерево отрезков, декартово дерево, система непересекающихся множеств);

— ключевые алгоритмы на графах (кратчайшие пути, минимальное остовное дерево, потоки, паросочетания);

— строковые алгоритмы (Z- и префикс-функция, суффиксный массив, бор);

— эвристические методы перебора (метод отжига).

уметь:

— анализировать задачу и выбирать оптимальную структуру данных или алгоритм для ее решения;

— оценивать временную и пространственную сложность предлагаемого решения;

— реализовывать изученные алгоритмы и структуры данных "с нуля";

— применять техники динамического программирования и предподсчета для оптимизации решения.

владеть:

— навыками решения сложных алгоритмических задач, характерных для соревнований и технических собеседований;

— методами отладки и тестирования алгоритмов.

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 аппарата, связанного с проектированием, разработкой, реализацией и оценкой качества программных продуктов и программных комплексов в различных областях человеческой деятельности
ОПК-6.	Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения	ОПК-6.1.	Знает алгоритмы разработки, компьютерные программы, а также алгоритмы вычислительной математики
		ОПК-6.2.	Умеет разрабатывать математические программные продукты и комплексы с использованием современных технологий программирования
		ОПК-6.3.	Имеет практический опыт разработки интеллектуальных информационных систем для визуализации результатов исследований
ПК-3.	Способен использовать методы математического и алгоритмического моделирования при решении теоретических и прикладных задач	ПК-3.1.	Знает основные методы математического и алгоритмического моделирования, а также их применение для решения теоретических и прикладных задач
		ПК-3.2.	Умеет разрабатывать и применять математические модели и алгоритмы для решения различных задач, анализируя полученные результаты
		ПК-3.3.	Имеет практический опыт использования методов математического и алгоритмического моделирования в реальных проектах или исследованиях

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	Графы	16	16	4	62	Домашнее задание
2	Деревья поиска	8	8	4	32	Домашнее задание Контрольная работа Коллоквиум
3	Строки	6	6		24	Домашнее задание
	Экзамен			**4**		
	Итого	***30***	***30***	***12***	***118***	
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	Графы	Графы в памяти компьютера. Способы храния. Базовые алгоритмы обхода и задачи на обходы. Алгоритмы на основе алгоритмов обхода. Обход в глубину. Мосты. Точки сочленения. Топологическая сортировка. Алгоритм Дейкстры. Алгоритм Форда-Беллмана. Алгоритм Левита. Алгоритм Флойда. Задача объединить-найти. Система не пересекающихся множеств. Алгоритм Краскала. Алгоритм Прима. Задача нахождения максимального потока в транспортной сети. Алгоритмы: Форда-Фалкерсона, Эдмондса — Карпа. Задача нахождения максимального потока в транспортной сети. Нахождение максимального паросочетания в двудольном графе: алгоритм Куна
2	Деревья поиска	Деревья поиска. Добавление, удаление элемента. Сбалансированные деревья поиска. АВЛ - дерево. Splay - дерево. B-дерево. Декартово дерево.
3	Строки	Строки. Z-функция. Префикс функция. Суффиксный массив. Бор. Задачи на Бор. Ахо-Корасик

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 2. Продвинутый уровень» в рамках текущего контроля успеваемости используются такие виды учебной работы, как лекция, семинары и домашние задания, коллоквиум, 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.

Аудиторная работа (бонусная активность) – активная работа студента на семинаре, его ответы на вопросы преподавателя и участие в дискуссии.

Для успешного участия в семинаре студентам рекомендуется заранее ознакомиться с темой обсуждения, прочитать необходимые материалы и подготовить вопросы. Важно активно слушать и вовлекаться в дискуссию, высказывая свои мнения и аргументируя их. При ответах на вопросы преподавателя стоит быть уверенным, четким и логичным, опираясь на изученный материал. Также полезно поддерживать диалог с однокурсниками, чтобы обогатить обсуждение и расширить свои знания.

Консультации – структурированные встречи, на которых преподаватели предоставляют индивидуальную или групповую помощь в освоении учебного материала, обсуждении вопросов и решении проблем, возникающих в процессе обучения.

Консультации могут включать разъяснение сложных тем, подготовку к экзаменам и помощь в выполнении проектных работ, что способствует более глубокому пониманию предмета и улучшению академической успеваемости.

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.

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.

Бонусные баллы – это оценки, которые студенты могут получить за выполнение дополнительных заданий.

Формат бонусных баллов позволяет студентам улучшить общую оценку по дисциплине (модулю) и стимулирует углубленное изучение материала.

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 2. 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	

Десятибалльная оценка	Пятибалльная оценка	Общая характеристика результата обучения по дисциплине (модулю)
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 2. Продвинутый уровень» оценивается следующим образом:

Активность	Вес	Количество	Описание
Домашние задания	30%	11	Набор задач по темам недели
Контрольная работа	20%	1	Письменная работа с набором задач, которые нужно решить за ограниченное время
Коллоквиум	30%	1	Устные ответы на вопросы, список которых известен студенту заранее
Экзамен	20%	1	Письменная или устная работа над заданием, направленным на проверку полученных знаний и навыков по дисциплине (модулю)

В рамках изучения дисциплины (модуля) возможно получение бонусных баллов.

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

Задание 1. Опишите способы хранения графа в памяти компьютера. Реализуйте представление графа с помощью списка смежности и матрицы смежности. Укажите преимущества и недостатки каждого из методов.

Задание 2. Напишите алгоритм обхода графа в глубину (DFS). Объясните, как он работает, и приведите пример обхода для конкретного графа. Укажите, как с его помощью можно определить, является ли граф связным.

Задание 3. Реализуйте алгоритм поиска в ширину (BFS) для невзвешенного графа. Напишите программу, которая выводит порядок посещения вершин и расстояния от стартовой вершины до всех остальных.

Задание 4. Примените алгоритм Дейкстры для поиска кратчайших путей от начальной вершины до всех остальных в взвешенном графе. Приведите пошаговое выполнение алгоритма на примере графа с 5 вершинами.

Задание 5. Напишите алгоритм для поиска мостов в неориентированном графе с использованием DFS. Объясните, как определяется, является ли ребро мостом, и приведите пример работы на конкретном графе.

Задание 6. Реализуйте систему непересекающихся множеств (DSU) с операциями объединения и поиска. Используйте эвристики сжатия путей и объединения по рангу. Продемонстрируйте работу структуры на последовательности операций.

Задание 7. Опишите и реализуйте алгоритм Краскала для построения минимального остовного дерева. Объясните, как используется DSU, и проанализируйте временную сложность алгоритма.

Задание 8. Напишите алгоритм для нахождения максимального паросочетания в двудольном графе с помощью алгоритма Куна. Приведите пример графа и покажите, как происходит построение паросочетания на каждом шаге.

Задание 9. Вычислите Z-функцию для строки "abacabadaba". Опишите пошагово, как заполняются значения функции, и объясните, как она может использоваться для поиска подстроки в строке.

Задание 10. Реализуйте структуру данных «бор» (trie) для хранения набора строк. Добавьте операции вставки строки и поиска по префиксу. Приведите пример использования бора для автодополнения в поисковой системе.

Примерные домашние задания

Домашнее задание 1.

1. Что такое алгоритм Евклида? Объясните, как он используется для нахождения НОД (наибольшего общего делителя) двух чисел.
2. Опишите, как работает решето Эратосфена для нахождения простых чисел. Какова его временная сложность?
3. Приведите примеры методов факторизации чисел и обсудите их эффективность.
4. Что такое метод двух указателей? Приведите пример его применения на практике.
5. Объясните, какие основные структуры данных вы знаете, и приведите примеры их использования в алгоритмах.

Домашнее задание 2.

1. Реализуйте структуру данных для представления графа с использованием списка смежности и матрицы смежности. Напишите функции для добавления и удаления вершин и ребер, а также для вывода графа на экран.
2. Реализуйте алгоритмы обхода графа в глубину (DFS) и в ширину (BFS). Напишите программу, которая выводит порядок посещения вершин для заданного графа и начальной вершины. Проанализируйте временную сложность каждого алгоритма.
3. Реализуйте алгоритм Дейкстры для поиска кратчайшего пути в взвешенном графе без отрицательных ребер. Напишите программу, которая находит кратчайший путь от заданной начальной вершины до всех остальных вершин графа.
4. Напишите программу для обнаружения циклов в ориентированном и неориентированном графах. Используйте алгоритм обхода в глубину с отслеживанием состояния вершин. Проверьте работу программы на нескольких примерах.
5. Реализуйте алгоритмы Крускала и Прима для поиска минимального остовного дерева в связном взвешенном неориентированном графе. Сравните эффективность обоих алгоритмов на различных тестовых данных.

Домашнее задание 3.

1. Реализуйте простую хеш-таблицу с методом цепочек для обработки коллизий. Поддержите операции: вставка, удаление и поиск ключа. Используйте встроенную хеш-функцию языка (например, hash() в Python) или реализуйте собственную для строк. Протестируйте работу на наборе строк и целых чисел.
2. Реализуйте хеш-функцию по методу Горнера для строк с использованием простого модуля (например, 1009 или 9973). Сравните количество коллизий при вставке списка слов (например, из текста или словаря) с использованием разных модулей и коэффициентов. Сделайте вывод о влиянии параметров на равномерность распределения.
3. Напишите программу, которая находит все подстроки заданной строки, которые встречаются более одного раза, с помощью полиномиального хеширования. Используйте

метод скользящего окна и хеш для сравнения подстрок за O(1). Проанализируйте сложность алгоритма.

4. Реализуйте структуру данных "множество строк" на основе хеш-таблицы. Добавьте методы: add(s), contains(s), remove(s). Сравните её производительность с использованием встроенного множества языка на больших объёмах данных (10 000–100 000 элементов).

5. Напишите отчёт (1 страница), в котором объясните:

— Как работает хэширование и зачем оно нужно в реальных приложениях

— Где используется: базы данных, кэширование, криптография, поисковые системы

— Какие проблемы возникают при плохом выборе хеш-функции (коллизии, атаки)

— Почему хеш-таблицы обеспечивают среднюю сложность O(1) для операций.

Задания для промежуточной аттестации по дисциплине (модулю)

№ п/п	Задание	Ответ	Компетенция
1	Какое ограничение на веса рёбер существует для корректной работы алгоритма Дейкстры?	неотрицательные веса	ОПК-4
2	Какова временная сложность алгоритма Форда-Беллмана для графа с V вершинами и E рёбрами?	O(V * E)	ОПК-4
3	Какую матрицу использует алгоритм Флойда-Уоршелла для хранения расстояний между всеми парами вершин?	матрица смежности / расстояний	ОПК-6
4	Как называется ребро, удаление которого увеличивает количество компонент связности графа?	мост	ОПК-6
5	Как называется вершина, удаление которой увеличивает количество компонент связности графа?	точка сочленения	ОПК-4
6	Какова асимптотическая сложность операций в системе непересекающихся множеств с эвристикой сжатия путей и рангами?	O(alpha(n)) / почти константная	ПК-3
7	Что утверждает теорема о максимальном потоке и минимальном разрезе в транспортной сети?	величина максимального потока равна пропускной способности минимального разреза	ПК-3
8	Для графов какого типа применяется алгоритм Куна для нахождения наибольшего паросочетания?	двудольные графы	ОПК-6
9	Какой тип графа необходим для выполнения топологической сортировки вершин?	ориентированный ациклический (DAG)	ОПК-4
10	Какова амортизированная сложность операции доступа к элементу в Splay-дереве?	O(log n)	ОПК-4

11	Какое свойство сохраняет Декартово дерево (Treap) относительно приоритетов узлов?	свойство кучи	ОПК-6
12	Для хранения данных на каких устройствах преимущественно используются B-деревья?	диски / внешняя память	ОПК-6
13	При каком значении фактора баланса в AVL-дереве требуется выполнение ротации?	2 или -2 (по модулю > 1)	ОПК-4
14	Для решения какой задачи предназначен алгоритм Ахо-Корасик?	поиск множества образцов в тексте	ПК-3
15	Какова сложность построения суффиксного массива простым сортированием всех суффиксов строки?	O(n^2 log n)	ОПК-4
16	Чему равно значение Z-функции в нулевой позиции для строки длины n?	n	ОПК-4
17	Какой алгоритм поиска подстроки использует префикс-функцию для работы за линейное время?	Кнута-Морриса-Пратта (KMP)	ОПК-6
18	От чего зависит объем памяти, занимаемый бором (trie) для набора строк?	от общего количества символов	ПК-3
19	Какой критерий выбора увеличивающего пути использует алгоритм Эдмондса-Карпа?	кратчайший по количеству рёбер (BFS)	ОПК-6
20	Какую структуру данных использует алгоритм Левита для хранения вершин в процессе работы?	очередь с приоритетами / дек	ОПК-4