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»

Направление подготовки: *02.04.01 Математика и компьютерные науки*

Направленность (профиль) подготовки: *Продуктовая аналитика*

Квалификация (степень) выпускника: *магистр*

Форма обучения: *очная*

Срок освоения программы: *2 года*

Год набора: *2025*

Москва
2025

ДОКУМЕНТ ПОДПИСАН
ЭЛЕКТРОННОЙ ПОДПИСЬЮ

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

Электронный документ

Содержание

1. **Краткая характеристика дисциплины (модуля)** 3
2. **Перечень планируемых результатов обучения** 4
3. **Тематический план** 6
4. **Содержание дисциплины (модуля)** 6
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» составлена в соответствии с федеральным государственным образовательным стандартом высшего образования – магистратура по специальности 02.04.01 Математика и компьютерные науки, профиль Продуктовая аналитика, утвержденный приказом Министерства науки и высшего образования Российской Федерации № 810 от 23.08.2017 года.

Изучение дисциплины (модуля) «Алгоритмы и структуры данных. Часть 1» является ключевым для формирования эффективного мышления программиста, позволяя оптимизировать решения задач, снижать затраты ресурсов и улучшать производительность программ в реальных приложениях.

Место дисциплины (модуля) в структуре образовательной программы

Настоящая дисциплина (модуль) включена в учебный план по программе подготовки магистратуры по направлению 02.04.01 Математика и компьютерные науки, профиль Продуктовая аналитика и входит в вариативную часть Блока 1, формируемую участниками образовательных отношений, как дисциплина по выбору.

Дисциплина (модуль) изучается на 1 или 2 курсе во 2, 3 или 4 семестре на выбор.

Цель изучения дисциплины (модуля): освоение фундаментальных принципов и методов анализа алгоритмов и структур данных для создания оптимальных и масштабируемых программных решений.

Задачи изучения дисциплины (модуля):

- познакомиться с разнообразием способов организации и управления данными в программных системах;
- изучить методы упорядочивания информации с учётом различных требований к производительности;
- рассмотреть техники минимизации ошибок и конфликтов при работе с данными в вычислительных структурах;
- освоить практические навыки применения встроенных средств программирования для решения сложных вычислительных задач;
- научиться проводить сравнительный анализ алгоритмических подходов с целью выбора наиболее эффективного решения.

В результате освоения дисциплины (модуля) обучающийся должен:

знать:

- принципы работы контейнеров в стандартной библиотеке любого языка программирования, в частности, Java;
- различные варианты решения задачи о сортировке;
- виды разрешения коллизий.

уметь:

- использовать подходящие инструменты из стандартной библиотеки языка программирования для реализации алгоритмов;
- оценивать сложность и время работы произвольного алгоритма.

владеть:

- навыками оценки различных существующих решений алгоритмических задач и, при необходимости, их улучшения;
- навыками выбора подходящих оптимальных алгоритмов для решения прикладных задач.

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)
УК-6.	Способен определять и реализовывать приоритеты собственной деятельности и способы ее совершенствования на основе самооценки	УК-6.1.	Знает основные методы самооценки и анализа своей деятельности, а также принципы управления временем и целеполагания
		УК-6.2.	Умеет ставить реалистичные и достижимые цели, определять приоритеты в своей деятельности, а также разрабатывать и внедрять планы по совершенствованию своих навыков и компетенций на основе полученной самооценки
		УК-6.3.	Имеет практический опыт применения методов самооценки в своей профессиональной деятельности, включая участие в тренингах, семинарах и проектах, направленных на развитие личной эффективности и профессионального роста
ОПК-2.	Способен создавать и исследовать новые математические модели в естественных науках, совершенствовать и разрабатывать концепции, теории и методы	ОПК-2.1.	Знает основные математические модели и методы, используемые в естественных науках, включая статистическое моделирование, дифференциальные уравнения и численные методы, а также современные подходы к исследованию и анализу данных
		ОПК-2.2.	Умеет разрабатывать и адаптировать математические модели для решения конкретных проблем в естественных науках, проводить их анализ и верификацию, а также интерпретировать полученные результаты в контексте научных исследований
		ОПК-2.3.	Имеет практический опыт создания и исследования математических моделей в

			рамках научных проектов или исследований, включая участие в публикациях, конференциях или коллаборациях, где были разработаны и апробированы новые концепции и методы
ПК-3.	Способен решать задачи профессиональной деятельности в области продуктовой аналитики, формулировать результаты анализа и выявлять последствия полученных данных для принятия обоснованных решений и оптимизации продуктов	ПК-3.1.	Знает методы и инструменты продуктовой аналитики
		ПК-3.2.	Умеет применять аналитические инструменты и программное обеспечение для обработки и визуализации данных, а также формулировать выводы на основе проведенного анализа
		ПК-3.3.	Имеет опыт работы над реальными проектами в области продуктовой аналитики, включая анализ пользовательского поведения и оптимизацию продуктов на основе полученных данных
ПК-4.	Способен публично представлять собственные и известные научные результаты	ПК-4.1.	Знает основные принципы эффективного публичного выступления, методы визуализации данных и основные требования к научным презентациям, включая структуру и содержание
		ПК-4.2.	Умеет четко и логично формулировать свои научные результаты, адаптируя их для различных аудиторий, а также использовать визуальные средства для улучшения восприятия информации
		ПК-4.3.	Имеет практический опыт участия в научных конференциях, семинарах или других мероприятиях, где успешно представлял свои и известные научные результаты, получая обратную связь и взаимодействуя с аудиторией

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 (Практические занятия)			
1	Базовые алгоритмы	4	4		17	Домашнее задание
2	Задача сортировки	4	4		17	Домашнее задание
3	Динамическое программирование	4	4		17	Домашнее задание
4	Основы дискретной теории вероятностей	6	6		24	Домашнее задание
5	Амортизированное время. Линейные контейнеры	2	2		9	Домашнее задание
6	Вероятностные алгоритмы. Хеширование. Хеш-таблицы	6	6		25	Домашнее задание
7	Деревья поиска	4	4		17	Домашнее задание
	Зачет			***4***		Алгоритмическое интервью
	Итого:	***30***	***30***	***4***	***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	Базовые алгоритмы	О-нотация. Асимптотический анализ. Рекуррентные соотношения. *Мастер теорема. Бинарный поиск: по ответу, вещественный Префиксные суммы (многомерные), подход двух указателей
2	Задача сортировки	Теорема о сортировке. Процедура Merge, сортировка слиянием. Подсчет числа инверсий. Поразрядные сортировки. Куча, HeapSort. Построение кучи за линейное время.
3	Динамическое программирование	Динамическое программирование 2: по подотрезкам, рюкзак, *по подмножествам Динамическое программирование 1: одномерное, двумерное
4	Основы дискретной теории вероятностей	Основы комбинаторики Дискретная теория вероятности. Условная вероятность, формула Байеса, формула полной вероятности Матожидание и дисперсия
5	Амортизированное время. Линейные контейнеры	Амортизационный анализ. Динамически расширяющийся массив. Списки. Стек, Очередь, Двусторонняя очередь
6	Вероятностные алгоритмы. Хеширование. Хеш-таблицы	Вероятностная сложность. Quicksort, quick select. Понятие хеш-функции. Разрешение коллизий методом цепочек. Универсальные семейства хеш-функций. Ассоциативные контейнеры. k-независимые семейства хеш-функций. Алгоритм FKS. Фильтр Блума

		Разрешение коллизий методом открытой адресации. Идеальное хеширование. Хеширование кукушкой
7	Деревья поиска	Деревья поиска: базовые операции. Красно-черное дерево: теорема о высоте, вставка. Splay-дерево Декартово дерево поиска по явному и неявному ключам

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. Солтис, М. Введение в анализ алгоритмов : практическое руководство / М. Солтис ; пер. с анг. А. В. Логунова. - Москва : ДМК Пресс, 2019. - 279 с. - ISBN 978-5-97060-696-4. - Текст : электронный. - URL: https://znanium.ru/catalog/product/2201221.

2. Доуни, А. Алгоритмы и структуры данных. Извлечение информации на языке Java : практическое руководство / А. Доуни. - Санкт-Петербург : Питер, 2018. - 240 с. - (Серия «Бестселлеры O'Reilly»). - ISBN 978-5-4461-0572-4. - Текст : электронный. - URL: https://znanium.com/catalog/product/1760855.

Дополнительная литература:

1. 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.

2. Меджедович, Д. Алгоритмы и структуры для массивных наборов данных : практическое руководство / Д. Меджедович, Э. Тахирович ; пер. с англ. А. В. Логунова. – Москва : ДМК Пресс, 2024. - 342 с. – ISBN 978-5-93700-250-1. - Текст : электронный. - URL: https://znanium.ru/catalog/product/2205044.

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	Катастрофы, стихийные бедствия, аварии, эпидемии. Солнечная и геомагнитная активность. /ежедневный обзор	http://www.disasters.chat.ru
2	Каталог по безопасности жизнедеятельности	http://www.eun.chat.ru
3	Научная электронная библиотека elibrary.ru библиотека	https://elibrary.ru/defaultx.asp
4	База данных для IT-специалистов	https://habr.com
5	База данных ScienceDirect	https://www.sciencedirect.com
6	Официальный сайт Министерства науки и высшего образования Российской Федерации	https://minobrnauki.gov.ru/
7	Федеральный портал «Российское образование"	https://www.edu.ru/
8	Информационная система "Единое окно доступа к образовательным ресурсам"	http://window.edu.ru/
9	Единая коллекция цифровых образовательных ресурсов	http://school-collection.edu.ru/
10	Федеральный центр информационно - образовательных ресурсов	http://fcior.edu.ru/
11	Сайт различных плагинов	https://maven.apache.org/plugins/
12	Maven central repository - хранилище библиотек и фреймворков	https://mvnrepository.com/repos/central

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
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» в рамках текущего контроля успеваемости используются такие виды учебной работы, как лекции, семинары, домашние задания, алгоритмическое интервью, 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.

Домашнее задание – набор заданий 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.

Алгоритмическое интервью – это тип технического собеседования, который проверяет умение разработчика решать задачи на алгоритмы и структуры данных. Его цель – не столько проверить знание синтаксиса языка, сколько оценить мышление, способность

к оптимизации и работу с ограничениями.

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».

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

Промежуточная аттестация по дисциплине (модулю) осуществляется в форме ***зачета***, 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:

Десятибалльная оценка	**Пятибалльная оценка**	**Оценка за зачет**	**Общая характеристика результата обучения по дисциплине (модулю)**
10	Отлично	Зачтено	Студент полностью владеет знаниями, изложенными в рабочей программе, и глубоко осмысляет дисциплину (модуль). 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	Зачтено	
8	Отлично	Зачтено	
7	Хорошо	Зачтено	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
6	Хорошо	Зачтено	

Десятибалльная оценка	Пятибалльная оценка	Оценка за зачет	Общая характеристика результата обучения по дисциплине (модулю)
			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.
5	Удовлетворительно	Зачтено	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	Зачтено	
3	Не сдан	Не зачтено	Студент не овладел обязательным минимумом знаний по предмету и не может ответить на вопросы, даже если преподаватель задает дополнительные наводящие вопросы.
2	Не сдан	Не зачтено	
1	Не сдан	Не зачтено	

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

Активность	Вес	Описание
Домашние задания	60%	Оцениваются по критериям. Можно набрать максимум 10 баллов за каждое из заданий.
Зачет	40%	Алгоритмическое интервью

Формула расчёта итоговой оценки по дисциплине (модулю) «Алгоритмы и структуры данных. Часть 1»: «0,6 × среднее за домашние задания + 0,4 × зачет».

Текущий контроль успеваемости обучающихся по дисциплине (модулю)

Примерные домашние задания

Домашнее задание

Input file:	`стандартный ввод`
Output file:	`стандартный вывод`
Time limit:	1 секунда
Memory limit:	256 мегабайт

В данной задаче вам необходимо написать функцию с заданным прототипом:

bool binarySearch(int[] array, int begin, int end, int target);

Она возвращает $true$, если на полуинтервале $[begin, end)$ массива $array$ имеется значение $target$.

Input

В первой строке дано число N ($1 \le N \le 10^5$) — длина массива $array$. На второй строке идут N целых чисел в порядке неубывания. Гарантируется, что по модулю они не превосходят 10^9.

На третьей строке идет единственное число Q ($1 \le Q \le 10^5$) — число вызовов функции $binarySearch$.

Далее идет Q строк запросов в формате «i j t» ($0 \le i < j \le N, |t| \le 10^9$): i, j — индексы массива, задающие полуинтервал для вызова функции, t — аргумент $target$ в вызове функции.

Output

Для каждого вызова функции выведите `YES`, если она вернула $true$, и `NO` — иначе.

Example

стандартный ввод	стандартный вывод
5 1 2 3 4 5 5 0 3 4 0 3 3 1 5 6 1 2 2 0 5 5	NO YES NO YES YES

Домашнее задание

Имя входного файла:	`стандартный ввод`
Имя выходного файла:	`стандартный вывод`
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Нужно отсортировать N рациональных чисел в порядке неубывания.

Формат входных данных

В первой строке идет число N ($1 \leqslant N \leqslant 10^5$), далее идет N строк из пар чисел p, q. Гарантируется, что ($-10^9 \leqslant p \leqslant 10^9, 1 \leqslant q \leqslant 10^9$). Считайте, что p — числитель, а q — знаменатель.

Формат выходных данных

Вывести отсортированный массив рациональных чисел. Каждое число должно быть несократимо, то есть НОД$(|p|, q) = 1$, q должно быть положительным.

Пример

стандартный ввод	стандартный вывод
6 1 2 2 4 2 1 0 3 -1 2 -1 3	-1 2 -1 3 0 1 1 2 1 2 2 1

Замечание

Например, пара $(0, 2)$ при выводе должна стать парой $(0, 1)$.

Домашнее задание

Имя входного файла:	`стандартный ввод`
Имя выходного файла:	`стандартный вывод`
Ограничение по времени:	1.5 секунд
Ограничение по памяти:	256 мегабайт

Вам дан словарь, состоящий из пар слов. Каждое слово является синонимом к парному ему слову. Все слова в словаре различны. Для каждого данного слова определите его синоним.

Формат входных данных

Программа получает на вход количество пар синонимов N ($0 \leqslant N \leqslant 10^5$). Далее следует N строк, каждая строка содержит ровно два слова-синонима.

Затем идет число Q ($1 \leqslant Q \leqslant 10^5$) — количество запросов к словарю. Далее на каждой следующей из Q строк идет слово, к которому надо вывести синоним.

Формат выходных данных

Программа должна вывести синонимы к данным слову на отдельных строках.

Пример

стандартный ввод	стандартный вывод
3 car plane mouse cat base stream 3 plane stream base	car base stream

Замечание

Для решения этой задачи можно написать любую из следующих структур данных:

- AVL-дерево — 3 балла
- B-дерево, RB-дерево — 4 балла

Примерное описание и задачи к алгоритмическому интервью

1. ПРАВИЛА ПРОВЕДЕНИЯ ЗАЧЕТА

Зачет проходит в формате алгоритмического интервью. Будут предложены две задачи: одна первого уровня и одна второго из списков ниже.

Решение задач будет представлять из себя лайвкодинг на своем ноутбуке. Будут проверяться работоспособность решения, его характеристики (время работы и потребляемая память), качество кода и ответы на сопутствующие вопросы по решению задачи.

Время проведения зачета составляет 45 минут без времени на подготовку. По истечении 45 минут ответ на зачет заканчивается и выставляется набранная на текущий момент оценка.

Список задач

1. ЗАДАЧИ 1-ГО УРОВНЯ

ЗАДАЧА 1

Дан массив $a_1, \ldots, a_n$ целых чисел. Необходимо найти максимальный по длине монотонный (строго возрастающий или строго убывающий) подотрезок (из подряд идущих элементов).

ЗАДАЧА 2

Места в кинотеатре расположены в один ряд. Только что пришедший зритель выбирает место, чтобы сидеть максимально далеко от остальных зрителей в ряду. То есть расстояние от того места, куда сядет зритель до ближайшего к нему зрителя должно быть максимально.

Гарантируется, что в ряду всегда есть свободные места и уже сидит хотя бы один зритель.

Напишите функцию, которая по заданному ряду мест (массиву из нулей и единиц) вернёт расстояние от выбранного места до ближайшего зрителя.

ЗАДАЧА 3

Дан отсортированный по неубыванию массив целых чисел a, индекс элемента $index$ и целое число k. Необходимо вернуть в любом порядке k чисел из массива, которые являются ближайшими по значению к элементу $a[index]$.

ЗАДАЧА 4

Отсортированный массив $a_1, \ldots, a_n$ циклически сдвинули вправо на k, то есть на вход подается массив $a_{n-k+1}, a_{n-k+2}, \ldots, a_n, a_1, \ldots, a_{n-k}$. Найдите k.

ЗАДАЧА 5

Дан отсортированный массив различных целых чисел $a_1, \ldots, a_n$ и число $K \geqslant 0$. Нужно найти количество пар элементов, разница которых $\leqslant K$.

ЗАДАЧА 6

Дан непустой массив из нулей и единиц. Нужно определить, какой максимальный по длине подынтервал единиц можно получить, удалив (пропустив) ровно один элемент массива. Вернуть 0, если такого подынтервала не существует.

Удалять один элемент из массива обязательно.

ЗАДАЧА 7

Напишите функцию, которая сдвигает все нули в массиве в конец.

ЗАДАЧА 8

Нужно реализовать функцию `distNoGreaterThanOne`, проверяющую, можно ли одну строку получить из другой не более, чем за одно исправление (удаление, добавление, изменение символа).

ЗАДАЧА 9

Даны две отсортированных по неубыванию последовательности целых чисел.

Необходимо вернуть все элементы из первой последовательности, которых нет во второй.

ЗАДАЧА 10

Даны две строки. Надо проверить, является ли первая подстрока подпоследовательностью (элементы необязательно идут подряд) второй.

ЗАДАЧА 11

На входе дана непустая строка. Требуется выяснить, можно ли удалить из нее ровно один символ так, чтобы получился палиндром.

ЗАДАЧА 12

Дан отсортированный массив $a_1, \ldots, a_n$. Постройте отсортированный массив из модулей элементов массива $\{|a_i|\}_{i=1}^{n}$.

ЗАДАЧА 13

Дан односвязный список. Необходимо проверить его на палиндромность.

ЗАДАЧА 14

Дан односвязный список. Сдвиньте его циклически на K вправо.

ЗАДАЧА 15

Даны два отсортированных односвязных списка. Верните объединение этих списков как отсортированный список.

ЗАДАЧА 16

Дан массив чисел $a_1, \ldots, a_{2n+1}$, в нем каждое число встречается ровно два раза кроме одного элемента. Найдите этот элемент.

2. ЗАДАЧИ 2-ГО УРОВНЯ

ЗАДАЧА 17

Дан массив различных целых чисел $a_1, \ldots, a_n$ и число $K \geqslant 0$. Нужно найти количество пар элементов, разница которых $\geqslant K$.

ЗАДАЧА 18

Даны даты заезда и отъезда каждого гостя. Для каждого гостя дата заезда строго раньше даты отъезда (то есть каждый гость останавливается хотя бы на одну ночь). В пределах одного дня считается, что сначала старые гости выезжают, а затем въезжают новые.

Найти максимальное число постояльцев, которые одновременно проживали в гостинице (считаем, что измерение количества постояльцев происходит в конце дня).

ЗАДАЧА 19

Даны 2 вектора целых чисел одинаковой длины, заданные в сжатой форме списками пар вида `(value, count)`. Например, вектор `[4, 4, 5]` задается как `[(4, 2), (5, 1)]`.

Необходимо посчитать скалярное произведение заданных векторов.

ЗАДАЧА 20

Дана строка S. Найти число подстрок (элементы идут подряд) без повторяющихся элементов.

ЗАДАЧА 21

Даны строки S и T. Найти любую подстроку S, являющуюся анаграммой T.

ЗАДАЧА 22

Дан массив $a_1, \ldots, a_n$. Найти минимальный по длине подотрезок с суммой равной K.

ЗАДАЧА 23

Дан массив $a_1, \ldots, a_n$. Найти минимальный по длине подотрезок с суммой кратной K.

ЗАДАЧА 24

Дан массив $a_1, \ldots, a_n$. Найти максимальную по длине подпоследовательность, состоящую из подряд идущих элементов. В качестве ответа выдать ее индексы.

Например, для `[1, 3, 2, 4, 3]` ответ `[0, 2, 4]`.

ЗАДАЧА 25

Дан массив $a_1, \ldots, a_n$. Необходимо проверить, существует ли такой индекс m, что множество (то есть набор уникальных элементов) из элементов $\{a_1, \ldots, a_{m-1}\}$ совпадает с множеством $\{a_m, \ldots, a_n\}$.

Например, для $[1, 2, 4, 3, 2, 2, 4, 3, 1, 2]$ подойдет граница $m = 4$ в 0-индексации.

ЗАДАЧА 26

Дан массив $a_1, \ldots, a_n$. Найдите количество пар индексов $i < j$ таких, что $a_i + a_j$ делится на K.

ЗАДАЧА 27

Дана строка S и Q запросов (l_i, r_i). Необходимо проверить для каждого из Q запросов, является ли $S[l_i : r_i]$ анаграммой какого-нибудь палиндрома.

ЗАДАЧА 28

Дан отсортированный массив $a_1, \ldots, a_n$. Найти $k < n$ пар элементов $a_i < a_j$ таких, что $|a_i - a_j|$ минимально.

Задания для промежуточной аттестации по дисциплине (модулю)

№ п/п	Задание	Ответ	Компетенция
1.	Выберете из нижеперечисленных контейнеров ассоциативные: А. Deque Б. PriorityQueue В. HashMap Г. ArrayList	В	ПК-3
2.	Какой из нижеперечисленных контейнеров позволяет проверять наличие элемента в нем за константное в среднем время: А. HashSet Б. TreeSet В. ArrayList Г. PriorityQueue	А	ПК-3
3.	Выберите из нижеперечисленных алгоритмов сортировки тот, который работает за O(N log N) в худшем случае А. HeapSort Б. BubbleSort В. InsertionSort	А	ПК-3
4.	Рассмотрим задачу проверки наличия элемента в неизменяемом наборе элементов. Алгоритм А: отсортируем набор с помощью сортировки слиянием, запрос проверки наличия элемента в наборе осуществим бинарным поиском Алгоритм Б: построим TreeSet на элементах, запрос проверки осуществим стандартными методами TreeSet Алгоритм В: построим HashSet на элементах, запрос проверки осуществим стандартными методами HashSet. Для каждого из алгоритмов выберите один из трех вариантов времени подготовки данных в О-нотации (*ответ запишите в виде последовательности чисел, где 121 означает, что для алгоритма А выбран первый вариант, для алгоритма Б - второй, для алгоритма В - первый*). 1. O(N) 2. O(N) в среднем 3. O(N log N)	332	ПК-3
5.	Рассмотрим задачу проверки наличия элемента в неизменяемом наборе элементов. Алгоритм: построим TreeSet на элементах, запрос проверки осуществим стандартными запросами TreeSet Какая дополнительная память потребуется, если N - число элементов в контейнере? (*ответ запишите в О-нотации)*.	O(N)	ПК-3
6.	Рассмотрим семейство ассоциативных контейнеров, построенных на основе хеш-таблиц. Какое матожидание времени проверки наличия элемента при использовании метода цепочек для разрешения коллизий с универсальным семейством хеш-	O(1)/O(1) в среднем	ОПК-2

	функций, если N - число элементов в контейнере. *Ответ запишите в О-нотации*		
7.	Рассмотрим семейство ассоциативных контейнеров, построенных на основе самобалансирующихся деревьев поиска. Какое время проверки наличия элемента, если N - число элементов в контейнере. *Ответ запишите в О-нотации*	O(log N)	ОПК-2
8.	Какой высоты в худшем случае может достичь бинарное дерево поиска без балансировки? *Ответ запишите в О-нотации*	O(N)	ОПК-2
9.	Какой из методов приоритизации задач позволяет наиболее эффективно управлять временем?	Матрица Эйзенхауэра	УК-6
10.	Какой тип данных лучше всего подходит для визуализации временных затрат алгоритмов?	Столбчатая диаграмма	ПК-4
11.	Назовите метод самооценки асимптотической сложности базового алгоритма.	О-нотация	УК-6
12.	Укажите способ определения приоритетов в выборе метода сортировки.	Время выполнения	УК-6
13.	Назовите технику совершенствования динамического программирования на основе самооценки.	Оптимизация состояний	УК-6
14.	Назовите подход к созданию математической модели в дискретной теории вероятностей.	Формула Байеса	ОПК-2
15.	Укажите метод исследования вероятностной сложности алгоритма Quicksort.	Монте-Карло симуляция	ОПК-2
16.	Назовите формат презентации результатов построения кучи в алгоритмах.	Слайды	ПК-4
17.	Укажите способ публичного представления модели динамического программирования.	Диаграмма состояний	ПК-4
18.	Назовите метод демонстрации работы хеш-таблицы с разрешением коллизий.	Видео-симуляция	ПК-4
19.	Укажите элемент эффективной презентации операций в деревьях поиска.	Визуализация баланса	ПК-4