

Приложение № 1
к приказу ректора
АНО ВО «Центральный университет»
от «01» сентября 2025 г. № 0901.64

**Автономная некоммерческая организация высшего образования
«Центральный университет»**

УТВЕРЖДАЮ

Ректор АНО ВО «Центральный
университет»

Е.В. Ивашкевич

**ДОПОЛНИТЕЛЬНАЯ ПРОФЕССИОНАЛЬНАЯ ПРОГРАММА –
ПРОГРАММА ПОВЫШЕНИЯ КВАЛИФИКАЦИИ
«АЛГОРИТМЫ И СТРУКТУРЫ ДАННЫХ ДЛЯ ПРЕПОДАВАТЕЛЕЙ»**

Трудоемкость обучения: 56 ак. часов

Москва

2025

1. ОБЩАЯ ХАРАКТЕРИСТИКА ПРОГРАММЫ

1.1. Общие положения

Настоящая дополнительная профессиональная программа – программа повышения квалификации «Алгоритмы и структуры данных для преподавателей» (далее – программа повышения квалификации) разработана на основании Федерального закона от 29 декабря 2012 г. № 273-ФЗ «Об образовании в Российской Федерации», приказа Минобрнауки России от 01 июля 2013 г. № 499 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным профессиональным программам» и приказа Минобрнауки России от 11 октября 2023 г. № 1678 «Об утверждении Правил применения организациями, осуществляющими образовательную деятельность, электронного обучения, дистанционных образовательных технологий при реализации образовательных программ».

Программа повышения квалификации реализуется в Автономной некоммерческой организации высшего образования «Центральный университет» (далее – АНО ВО «Центральный университет»).

Разработчик программы:

- Шведова Мария Сергеева, координатор образовательных проектов отдела разработки образовательных программ ООО «ТЦР».

Программа повышения квалификации разработана в инициативном порядке.

Программа реализуется на русском языке.

1.2. Цель реализации программы

Программа повышения квалификации нацелена на совершенствование и (или) получение слушателем новой компетенции, необходимой для профессиональной деятельности, и (или) повышение профессионального уровня знаний в области алгоритмического мышления, анализа и реализации

алгоритмов, а также эффективного использования структур данных в задачах обработки и хранения информации.

1.3. Категории обучающихся

Основной категорией обучающихся, на которую рассчитана программа повышения квалификации, являются действующие преподаватели алгоритмов и структур данных в образовательных организациях высшего образования.

1.4. Требования к уровню подготовки поступающего на обучение, необходимому для освоения программы

К освоению программы допускаются лица, имеющие среднее профессиональное и (или) высшее образование, а также лица, получающие среднее профессиональное и (или) высшее образование. Наличие указанного образования или факта его получения должно подтверждаться документом государственного или установленного образца.

1.5. Критерии отбора на программу

К освоению программы допускаются лица, соответствующие следующим критериям:

- преподавание в прошедшем или следующем семестре дисциплины «Алгоритмы и структуры данных» или смежной с ней;
- преподавание в образовательной организации высшего образования.

Для проверки соответствия лиц, принимаемых на обучение, критериям допуска, поступающим на программу необходимо описать свой опыт преподавания и мотивацию участия в программе в сопроводительном письме, а также решить специальный контекст не менее, чем на 328 баллов.

1.6. Перечень нормативных документов, определяющих квалификационные требования к выпускнику программы

— Федеральный государственный образовательный стандарт по направлению подготовки (специальности) 44.03.01 «Педагогическое образование» и уровню высшего образования Бакалавриат, утвержденный приказом Минобрнауки России от 22 февраля 2018 г. № 121;

— Профессиональный стандарт «Программист», утвержденный приказом Министерства труда и социальной защиты Российской Федерации от 18 ноября 2013 г. № 679н.

1.7. Планируемые результаты обучения

а) Перечень профессиональных компетенций, совершенствование или получение которых осуществляется в результате обучения:

- Способен эффективно преподавать дисциплины, связанные с алгоритмами и структурами данных, в соответствии с требованиями ФГОС ВО и современными тенденциями IT-отрасли.

- Способен разрабатывать и актуализировать учебные материалы, лабораторные работы и задания по алгоритмической подготовке с учётом уровня подготовки студентов.

- Способен выбирать и объяснять оптимальные структуры данных и алгоритмы для решения типовых и сложных вычислительных задач.

- Способен организовать практико-ориентированное обучение, включая решение задач на программирование в стиле технических собеседований (LeetCode-подобные задачи).

- Способен использовать современные цифровые платформы и инструменты для преподавания алгоритмов (виртуальные тренажёры, визуализаторы, онлайн-песочницы).

- Способен оценивать уровень освоения алгоритмических компетенций обучающимися и корректировать методику преподавания на основе обратной связи.

б) Квалификационные требования к выпускнику программы:

Слушатель в результате освоения программы будет способен выполнять следующие должностные обязанности, приведённые в приказе Минздравсоцразвития от 11 января 2011 г. № 1н «Об утверждении единого квалификационного справочника должностей руководителей, специалистов и служащих, раздел «Квалификационные характеристики должностей руководителей и специалистов высшего профессионального и дополнительного профессионального образования».

Преподаватель

Организует и проводит учебную и учебно-методическую работу по всем видам учебных занятий, за исключением чтения лекций. Участвует в научно-исследовательской работе кафедры, иного подразделения образовательного учреждения. Обеспечивает выполнение учебных планов и программ. Под руководством профессора, доцента или старшего преподавателя разрабатывает или принимает участие в разработке методических пособий по видам проводимых занятий и учебной работы, организует и планирует методическое и техническое обеспечение учебных занятий. Создает условия для формирования у обучающихся (студентов, слушателей) основных составляющих компетентности, обеспечивающей успешность будущей профессиональной деятельности выпускников. Принимает участие в воспитательной работе с обучающимися (студентами, слушателями), в организации их научно-исследовательской работы, в профессиональной ориентации школьников, в разработке и осуществлении мероприятий по укреплению, развитию, обеспечению и совершенствованию материально-технической базы учебного процесса, обеспечению учебных подразделений и лабораторий оборудованием. Контролирует и проверяет выполнение обучающимися (студентами, слушателями) домашних заданий. Контролирует соблюдение обучающимися (студентами, слушателями) правил по охране труда и пожарной безопасности при проведении учебных занятий, выполнении лабораторных работ и практических занятий. Участвует в организуемых в рамках тематики

Электронный документ

направлений исследований кафедры семинарах, совещаниях и конференциях, иных мероприятиях образовательного учреждения.

в) Идентификаторы достижения компетенций:

Слушатель, освоивший программу повышения квалификации, должен:

знать:

— основные типы и классификацию алгоритмов (рекурсивные, жадные, динамическое программирование, «разделяй и властвуй»);

— принципы построения, функционирования и области применения ключевых структур данных: массивы, списки, стеки, очереди, хеш-таблицы, деревья, графы;

— методы анализа временной и пространственной сложности алгоритмов, включая использование O-нотации и оценку в лучшем, среднем и худшем случаях.

уметь:

— разрабатывать, оптимизировать и обосновывать выбор алгоритмов и структур данных при решении типовых и прикладных задач;

— преподавать сложные алгоритмические концепции доступно и наглядно, используя визуализации, примеры и практико-ориентированные задания;

— создавать и адаптировать учебные материалы, лабораторные работы и контрольные задания по дисциплине «Алгоритмы и структуры данных» для различных уровней подготовки студентов.

владеть:

— навыками применения современных цифровых образовательных платформ и инструментов (онлайн-песочницы, визуализаторы алгоритмов, системы автоматической проверки кода);

— методиками оценивания уровня усвоения алгоритмических компетенций обучающимися, включая критерии проверки решений задач программирования;

— профессиональной терминологией и подходами, используемыми в IT-индустрии, для обеспечения соответствия образовательного процесса актуальным отраслевым требованиям.

1.8. Трудоемкость программы

Нормативная трудоемкость обучения по данной программе – 56 академических часов, включая все виды контактной и самостоятельной работы слушателя.

1.9. Форма и сроки обучения

Обучение по программе повышения квалификации осуществляется в очной форме обучения с использованием дистанционных образовательных технологий и электронного обучения.

Формат обучения на программе – онлайн синхронный. Взаимодействие слушателей с преподавателями на занятии происходит в режиме реального времени посредством использования технических средств.

Минимальный срок обучения на программе составляет 3 месяца.

1.10. Режим занятий

Длительность одного занятия – 1-2 академических часа.

Для всех занятий академический час устанавливается продолжительностью 45 минут.

2. ПРОГРАММА УЧЕБНОГО КУРСА

2.1. Учебный план программы повышения квалификации «Алгоритмы и структуры данных для преподавателей»

Продолжительность обучения – 56 ак. часов.

Форма обучения – очная.

№ п/п	Наименование разделов учебного курса	Всего, академ. ч.	Контактная работа (академ. ч.)		Самост. работа (академ. ч.)	Формы аттестац ии и контроля знаний
			лекции	семинары (практич. занятия)		
1.	Наиболее частые алгоритмы и методы их решения	40	10	20	10	
2.	Современное преподавание	8	2	4	2	
	ВСЕГО:	48	12	24	12	
	Итоговая аттестация	8		8		<i>Зачет</i>
	ИТОГО:	56	12	32	12	

**2.2. Учебно-тематический план программы повышения
квалификации «Алгоритмы и структуры данных для
преподавателей»**

	Наименование разделов и тем	Всего, академ. ч.	Контактная работа (академ. ч.)		Самост. работа (академ. ч.)	Формы аттестации и контроля знаний
			лекции	семинары (практич. занятия)		
1.	Наиболее частые алгоритмы и методы их решения	40	10	20	10	
1.1.	Введение, сортировки, бинарный поиск	4	1	2	1	Подготовка к семинару, домашнее задание
1.2.	Базовые структуры данных	4	1	2	1	Подготовка к семинару, домашнее задание
1.3.	Деревья	4	1	2	1	Подготовка к семинару, домашнее задание
1.4.	Методы решения задач. Часть 1	4	1	2	1	Подготовка к семинару, домашнее задание
1.5.	Строки	4	1	2	1	Подготовка к семинару, домашнее задание
1.6.	Графы. Часть 1	4	1	2	1	Подготовка к семинару, домашнее задание
1.7.	Динамическое программирование	4	1	2	1	Подготовка к семинару, домашнее задание
1.8.	Дерево отрезков. Часть 1	4	1	2	1	Подготовка к семинару, домашнее задание
1.9.	Дерево отрезков. Часть 2	4	1	2	1	Подготовка к семинару, домашнее задание
1.10.	Методы решения задач. Часть 2	4	1	2	1	Подготовка к семинару, домашнее задание
2.	Современное преподавание	8	2	4	2	

2.1.	Проблемы и методологии преподавания	4	1	2	1	Подготовка к семинару, домашнее задание
2.2.	Организация тестирования на образовательной платформе	4	1	2	1	Подготовка к семинару, домашнее задание
	ВСЕГО:	48	12	24	12	
	Итоговая аттестация	8		8		<i>Зачет</i>
	ИТОГО:	56	12	32	12	

2.3. Календарный учебный график

Наименование раздела	Темы раздела	Академ. часов	Учебные месяцы		
			1	2	3
Наиболее частые алгоритмы и методы их решения	Введение, сортировки, бинарный поиск	4	4		
	Базовые структуры данных	4	4		
	Деревья	4	4		
	Методы решения задач. Часть 1	4	4		
	Строки	4		4	
	Графы. Часть 1	4		4	
	Динамическое программирование	4		4	
	Дерево отрезков. Часть 1	4		4	
	Дерево отрезков. Часть 2	4			4
	Методы решения задач. Часть 2	4			4
Современное преподавание	Проблемы и методологии преподавания	4			4
	Организация тестирования на образовательной платформе	4			4
Итоговая аттестация		8			8

3. РАБОЧАЯ ПРОГРАММА КУРСА

3.1. Содержание курса

№ п/п	Наименование раздела	Наименование темы	Содержание темы (тезисно)
1.	Наиболее частые алгоритмы и методы их решения	Введение, сортировки, бинарный поиск	Основы анализа сложности: $O(n)$, $O(\log n)$, $O(n^2)$. Сравнение сортировок: по скорости, памяти, устойчивости. Бинарный поиск: условия применения и реализация.
		Базовые структуры данных	Массивы, стек, очередь — операции и использование. Связный список: плюсы/минусы vs массив. Хеш-таблица: коллизии и способы их разрешения.
		Деревья	Бинарное дерево поиска: вставка, удаление, поиск. Обходы: in-order, pre-order, post-order. Сбалансированные деревья: идея балансировки (AVL, КЧ-деревья).
		Методы решения задач. Часть 1	Метод грубой силы — простота и неэффективность. «Разделяй и властвуй» — примеры: сортировка слиянием. Рекурсия: стек вызовов и базовый случай.
		Строки	Поиск подстроки: наивный алгоритм и КМП. Префикс-функция — основа оптимизации. Работа со строками в разных языках.
		Графы. Часть 1	Представление графа: матрица и список смежности. DFS и BFS — реализация и различия. Применение: связность, циклы, двудольность.
		Динамическое программирование	Основные принципы: подзадачи и запоминание. Bottom-up и top-down подходы. Задачи: Фибоначчи, рюкзак, НОП.
		Дерево отрезков. Часть 1	Назначение: запросы на диапазоне. Построение и хранение в массиве. Запрос и обновление за $O(\log n)$.
		Дерево отрезков. Часть 2	Ленивые вычисления — оптимизация обновлений. Применение: диапазонные изменения. Сравнение с деревом Фенвика и Sparse Table.
		Методы решения задач. Часть 2	Жадные алгоритмы — примеры и ограничения. Два указателя и скользящее окно. Бинарный поиск по ответу.
2.	Современное преподавание	Проблемы и методологии преподавания	Трудности студентов: абстракция, сложность анализа. Активные методы: парное программирование, кейсы. Визуализации и платформы (LeetCode, VisuAlgo) в обучении.

		Организация тестирования на образовательной платформе	Настройка типов заданий (тесты, код, открытые вопросы) и их интеграция в курс. Использование автоматической проверки для задач по алгоритмам и структурам данных. Настройка сроков, попыток, режимов тестирования и анализа результатов студентов.
--	--	---	--

3.2. Методические указания для обучающихся по освоению курса

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 удастся, 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 слушатели 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 задач по пройденным темам.

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 слушателей, направленная на углубленное изучение отдельных тем и вопросов курса.

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

4. ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИЕ УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ

4.1. Требования к кадровым условиям реализации программы

Реализация программы обеспечивается штатными руководящими и научно-педагогическими работниками АНО ВО «Центральный университет», а также внешними совместителями, работающими по договорам гражданско-правового характера. Научно-педагогические работники, осуществляющие преподавание данной программы, имеют образование, соответствующее профилю курса, или конкретный опыт реализации разработок и иной формы практической деятельности по направлению курса.

4.2. Требования к материально-техническим условиям реализации программы

Университет располагает материально-технической базой, соответствующей действующим противопожарным правилам и нормам и обеспечивающей проведение всех видов учебной деятельности, предусмотренных учебным планом.

Помещения 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 обеспечивается в учебных аудиториях, оснащенных:

- столами и стульями;
- компьютерной техникой;
- 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 к 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, 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

4.3. Учебно-методическое обеспечение программы

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

Каждый слушатель 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.

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

Основная литература:

1. Программирование: теоремы и задачи. – 2-е изд., испр. и доп. – М.: МЦНМО, 2004. – 296 с.

2. Андреева Е. В. Программирование — это так просто, программирование — это так сложно. Современный учебник программирования. – М.: МЦНМО, 2018. – 184 с.

Дополнительная литература:

1. Быкова, В. В. Комбинаторные алгоритмы: множества, графы, коды/Быкова В.В. - Краснояр.: СФУ, 2015. – 152 с.: ISBN 978-5-7638-3155-9. – Текст : электронный. – URL: <https://znanium.com/catalog/product/550333>.

5. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ

5.1. Формы контроля

В процессе изучения курса в рамках текущего контроля успеваемости используются такие виды учебной работы, как лекции, семинары, домашние задания, а также различные виды самостоятельной работы обучающихся по заданию преподавателя, направленные на развитие навыков профессиональной лексики, закрепление практических профессиональных компетенций, поощрение инициатив.

Итоговая аттестация по программе осуществляется в форме *зачета*, при этом проводится оценка компетенций, сформированных по курсу.

5.2. Система оценивания результатов обучения по курсу

Для оценивания текущего контроля успеваемости и итоговой 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 курсу
10	Отлично	Зачтено	Обучающийся полностью владеет знаниями, изложенными в рабочей программе, и глубоко осмысляет курс. 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. Обучающийся
9	Отлично	Зачтено	
8	Отлично	Зачтено	

Десятибалльная оценка	Пятибалльная оценка	Оценка за зачет	Общая характеристика результата обучения по курсу
			хорошо знаком с методами исследования, необходимыми для практической деятельности, и умеет связывать теоретические аспекты курса с практическими задачами.
7	Хорошо	Зачтено	Обучающийся 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. Обучающийся хорошо знаком с методами исследования, необходимыми для практической деятельности, и умеет связывать теоретические аспекты предмета с практическими задачами.
6	Хорошо	Зачтено	
5	Удовлетворительно	Зачтено	Обучающийся обладает базовыми знаниями по курсу, но испытывает трудности при самостоятельных ответах и использует неточные формулировки. В ходе ответов он допускает ошибки, касающиеся сути вопросов. Обучающийся способен решать только самые простые задачи и владеет лишь минимальным набором методов исследования.
4	Удовлетворительно	Зачтено	
3	Не сдан	Не зачтено	Обучающийся не овладел обязательным минимумом знаний по предмету и не может ответить на вопросы, даже если преподаватель
2	Не сдан	Не зачтено	
1	Не сдан	Не зачтено	

Десятибалльная оценка	Пятибалльная оценка	Оценка за зачет	Общая характеристика результата обучения по курсу
			задает дополнительные наводящие вопросы.

5.3. Примерные задания текущего контроля успеваемости

Домашнее задание по теме «Введение, сортировки, бинарный поиск»

Задача А. Двоичный поиск

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Реализуйте алгоритм бинарного поиска. Вам нужно ответить на несколько вопросов "Присутствует ли элемент x " в отсортированном массиве.

Формат входных данных

В первой строке содержатся числа n и k ($1 \leq n, k \leq 10^5$).

Во второй строке задаются n элементов первого массива, отсортированного по возрастанию, а в третьей строке — k вопросов. Все элементы целые, в диапазоне $[-10^9; 10^9]$.

Формат выходных данных

Для каждого из k чисел второго массива выведите в отдельную строку «YES», если это число встречается в первом массиве, и «NO» в противном случае.

Пример

стандартный ввод	стандартный вывод
10 5	NO
1 2 3 4 5 6 7 8 9 10	NO
-2 0 4 9 12	YES
	YES
	NO

Замечание

В массиве чисел от 1 до 10 есть числа 4, 9, но нет чисел -2, 0, 12

Домашнее задание по теме «Базовые структуры данных»

Задача А. Минимум на стеке

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Вам требуется реализовать структуру данных, выполняющую следующие операции:

1. Добавить элемент x в конец структуры.
2. Удалить последний элемент из структуры.
3. Выдать минимальный элемент в структуре.

Формат входных данных

В первой строке входного файла задано одно целое число n — количество операций ($1 \leq n \leq 10^6$). В следующих n строках заданы сами операции. В i -й строке число t_i — тип операции (1, если операция добавления, 2, если операция удаления, 3, если операция минимума). Если задана операция добавления, то через пробел записано целое число x — элемент, который следует добавить в структуру ($-10^9 \leq x \leq 10^9$). Гарантируется, что перед каждой операцией удаления или нахождения минимума структура не пуста.

Формат выходных данных

Для каждой операции нахождения минимума выведите одно число — минимальный элемент в структуре. Ответы разделяйте переводом строки.

Пример

стандартный ввод	стандартный вывод
8	-3
1 2	2
1 3	2
1 -3	
3	
2	
3	
2	
3	

Домашнее задание по теме «Деревья»

Задача А. Примитивы

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Дано дерево с корнем в вершине 0 (необязательно двоичное). Посчитайте:

1. Высоту дерева – максимальное расстояние от корня до листа.
2. Диаметр дерева – максимальная длина пути между двумя вершинами (естественно, путь не должен проходить через одну вершину несколько раз).
3. Для каждой вершины найдите её глубину – длину пути от корня до вершины.

Формат входных данных

В первой строке вводится натуральное число n – размер дерева ($2 \leq n \leq 10^5$). В следующей строке записано $n - 1$ целое число p_i – предок вершины i ($0 \leq p_i < i$).

Формат выходных данных

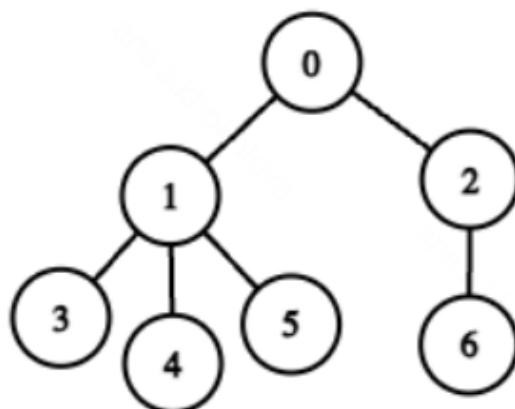
Выведите две строки. В первой строке выведите два числа: высоту и диаметр дерева. Во второй строке для каждой вершины выведите её глубину.

Примеры

стандартный ввод	стандартный вывод
7 0 0 1 1 1 2	2 4 0 1 1 2 2 2 2
6 0 1 2 2 2	3 3 0 1 2 3 3 3

Замечание

Дерево из первого примера:



Домашнее задание по теме «Методы решения задач. Часть 1»

Задача А. Суммы и XOR-ы на отрезках

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Задан числовой массив $a[1..n]$. Необходимо выполнить m операций вычисления суммы или хог на отрезке $[l, r]$.

Справка: операция хог – побитовое исключающее или – равно 0, если биты равны и 1 иначе. Существует во всех современных языках программирования, например, в языках C++, Java и Python она обозначена как « \wedge », в Pascal – как «хог».

Пример:

$$a = 01101000_2$$

$$b = 10101001_2$$

$$c = 11000001_2$$

где $c = a \wedge b$.

Формат входных данных

Первая строка входного файла содержит число n – размерность массива.

Во второй строке записаны n чисел – элементы массива.

Третья строка содержит число m – количество запросов суммы/хог. Следующие m строк содержат тройку чисел: $q_i - 1$, если запрос суммы, и 2, если запрос хог-а; l_i и r_i ($1 \leq l_i \leq r_i \leq n$), описывающие отрезки. Все числа во входных данных натуральные, не превосходящие $5 \cdot 10^5$.

Формат выходных данных

В выходной файл для каждого запроса на новой строке выведите результат суммы на отрезке для всех запросов типа 1 и хог-а – для запросов типа 2.

Пример

стандартный ввод	стандартный вывод
5	2
3 1 8 7 3	15
3	8
2 1 2	
1 3 4	
1 3 3	

Подготовка к семинару по теме «Строки»

1. Почему поиск подстроки «в лоб» (сравнивать каждый символ) может быть медленным? Приведите пример строки, где он будет работать очень долго.

2. Что такое префикс-функция? Как она помогает не начинать поиск с начала каждый раз, когда символы не совпали?

3. Построй префикс-функцию для строки "ABRACADABRA". Почему на 10-м месте она равна 1?

4. В Python строки нельзя менять, а в C++ – можно. Как это влияет на скорость, если много раз склеивать строки?

5. Вам нужно найти миллионы одинаковых фраз в логах сервера. Почему обычный `in` в Python может не хватить? Как вы бы ускорили поиск?

Подготовка к семинару по теме «Графы. Часть 1»

1. Почему для графа с 1000 вершинами и 5000 рёбер лучше использовать список смежности, а не таблицу?

2. Почему BFS находит самый короткий путь, а DFS – нет? Нарисуйте простой пример, где DFS найдёт путь длиной 4, а есть и путь длиной 2.

3. Как DFS узнаёт, что в графе есть цикл? Почему важно проверять, не возвращаешься ли ты по тому же ребру, по которому пришёл?

4. Что значит «граф двудольный»? Как с помощью BFS или DFS проверить, можно ли раскрасить его в два цвета?

5. Вы анализируете соцсеть: можно ли разделить пользователей на две группы, чтобы никто не дружил со своим же группой? Какую задачу вы решаете?

Подготовка к семинару по теме «Динамическое программирование»

1. Что значит «перекрывающиеся подзадачи»? Приведите пример из жизни – например, как приготовление блюда может повторять одни и те же шаги.

2. Почему в задаче про рюкзак вы должны заполнять таблицу по порядку – сначала предметы, потом веса? Что случится, если перепутать?

3. Чем отличается «сверху вниз» (рекурсия + кэш) от «снизу вверх» (таблица) в DP? Когда удобнее каждый из способов?

4. Почему вычисление числа Фибоначчи рекурсией — медленно, а с DP — быстро? Нарисуйте, как много раз повторяются одни и те же вызовы.

5. Вы хотите купить вещи на бюджет 1000 рублей, чтобы получить максимум пользы. Что это за задача? А если можно брать одну и ту же вещь много раз – как изменится решение?

Подготовка к семинару по теме «Дерево отрезков. Часть 1»

1. Зачем использовать дерево отрезков, если можно просто пройти по массиву и посчитать сумму? Когда оно реально помогает?

2. Почему для массива из 5 элементов выделяют 20 ячеек в дереве, а не 10? Что происходит, если число не степень двойки?

3. Как дерево отрезков быстро находит сумму на отрезке? Почему оно не смотрит все элементы, а «прыгает» по узлам?

4. Если вместо суммы нужно искать минимум, что нужно изменить в дереве отрезков? (Подсказка: вместо $+$ – \min)

5. Вы мониторите температуру: приходят тысячи данных, и нужно быстро спрашивать: «Какая была самая низкая температура с 10:00 до 14:00?» Или обновлять показание. Почему дерево отрезков — идеальный выбор?

5.4. Критерии оценки освоения программы повышения квалификации

Формат итоговой аттестации: решение итогового конкурса.

Примеры заданий:

Задача А. Шестерёнки

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 0.25 секунд
Ограничение по памяти: 256 мегабайт

Даны две сцепленные шестеренки. У одной шестеренки N зубцов, у другой – K . Требуется найти, какое минимальное число поворотов на один зубчик требуется сделать, чтобы шестеренки вернулись в исходное состояние.

Формат входных данных

В единственной строке – два натуральных числа N и K , не превосходящих 10 миллионов.

Формат выходных данных

Выведите искомое количество зубчиков. Гарантируется, что оно не более миллиарда.

Примеры

стандартный ввод	стандартный вывод
2 3	6
6 21	42

Задача В. Разложение на простые

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 0.25 секунд
Ограничение по памяти: 256 мегабайт

Требуется разложить целое число N на простые множители и вывести результат в порядке возрастания.

Формат входных данных

Программе дано число N ($2 \leq N \leq 10^9$).

Формат выходных данных

Вывести разложение N на простые множители.

Примеры

стандартный ввод	стандартный вывод
2	2
1008	2^4*3^2*7

Задача С. Гипотеза Гольдбаха

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	0.25 секунд
Ограничение по памяти:	256 мегабайт

Гипотеза Гольдбаха (не доказанная до сих пор) утверждает, что любое четное число (кроме 2) можно представить в виде суммы двух простых чисел. Проверьте её!

Формат входных данных

Программа получает на вход одно натуральное четное число n ($4 \leq n \leq 10^6$).

Формат выходных данных

Для заданного четного n выведите через пробел два простых числа p и q , таких, что $p \leq q$ и $p + q = n$. Так как таких разложений может быть несколько, выведите то, в котором p минимально.

Примеры

стандартный ввод	стандартный вывод
6	3 3
992	73 919

Задача D. Последняя цифра $N!$

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Требуется найти последнюю ненулевую цифру числа $N! = 1 \cdot 2 \cdot 3 \cdot \dots \cdot N$.

Формат входных данных

Входной файл содержит единственное натуральное число N ($N \leq 10^6$).

Формат выходных данных

В выходной файл выведите ответ на задачу.

Примеры

стандартный ввод	стандартный вывод
1	1
5	2

6. ВЫХОДНЫЕ ДОКУМЕНТЫ

Лицам, успешно освоившим соответствующую программу повышения квалификации и прошедшим итоговую аттестацию, выдаётся удостоверение о повышении квалификации. Удостоверение выдается на бланке, являющемся защищенной от подделок полиграфической продукцией, образец которого самостоятельно установлен образовательным учреждением.

Лицам, не прошедшим итоговую аттестацию или получившим на итоговой аттестации неудовлетворительные результаты, а также лицам, освоившим часть программы повышения квалификации и (или) отчисленным из организации, выдается справка об обучении или о периоде обучения, по образцу, самостоятельно устанавливаемому организацией.

При освоении программы параллельно с получением среднего профессионального образования и (или) высшего образования удостоверение о повышении квалификации выдается одновременно с получением соответствующего документа об образовании и о квалификации. На момент завершения программы лицам, получающим среднее профессиональное и (или) высшее образование, успешно освоившим соответствующую программу повышения квалификации и прошедшим итоговую аттестацию, выдается справка об обучении или о периоде обучения, по образцу, самостоятельно устанавливаемому организацией.