МАТЕРИАЛЫ ЗАДАНИЙ
Всероссийской междисциплинарной олимпиады школьников 8–11 класса «Национальная технологическая олимпиада»

по профилю «Ядерные технологии»

2024/25 учебный год

ntcontest.ru

УДК 373.5.016:621.039
ББК 74.263.0
Я34

Авторы:
К. М. Аспидов, С. Н. Деменков, О. В. Зубков, Н. Ю. Кузнецов, И. Г. Кулло, И. Б. Мамай

Я34 Всероссийская междисциплинарная олимпиада школьников 8–11 класса «Национальная технологическая олимпиада». Учебно-методическое пособие Том 23 **Ядерные технологии**
— М.: Ассоциация участников технологических кружков, 2025. — 188 с.

ISBN 978-5-908021-22-7

Данное пособие разработано коллективом авторов на основе опыта проведения всероссийской междисциплинарной олимпиады школьников 8–11 класса «Национальная технологическая олимпиада» в 2024/25 учебном году, а также многолетнего опыта проведения инженерных соревнований для школьников. В пособии собраны основные материалы, необходимые как для подготовки к олимпиаде, так и для углубления знаний и приобретения навыков решения инженерных задач.

В издании приведены варианты заданий по профилю Национальной технологической олимпиады за 2024/25 учебный год с ответами, подробными решениями и комментариями. Пособие адресовано учащимся 8–11 классов, абитуриентам, школьным учителям, наставникам и преподавателям учреждений дополнительного образования.

Методические материалы также могут быть полезны студентам и преподавателям направлений, относящихся к группам:
12.00.00 Фотоника, приборостроение, оптические и биотехнические системы и технологии
14.00.00 Ядерная энергетика и технологии
15.00.00 Машиностроение
27.00.00 Управление в технических системах ания, центров молодежного и инновационного творчества и детских технопарков.

ISBN 978-5-908021-22-7
9 785908 021227 >

УДК 373.5.016:621.039
ББК 74.263.0

Оглавление

1. Введение

1.1. Национальная технологическая олимпиада

Всероссийская междисциплинарная олимпиада школьников 8–11 класса «Национальная технологическая олимпиада» (далее — Олимпиада, НТО) проводится в соответствии с распоряжением Правительства Российской Федерации от 10.02.2022 № 211-р при координации Министерства науки и высшего образования Российской Федерации и при содействии Министерства просвещения Российской Федерации, Министерства цифрового развития, связи и массовых коммуникаций Российской Федерации, Министерства промышленности и торговли Российской Федерации, Ассоциации участников технологических кружков, Агентства стратегических инициатив по продвижению новых проектов, АНО «Россия — страна возможностей», АНО «Платформа Национальной технологической инициативы» и Российского движения детей и молодежи «Движение Первых».

Проектное управление Олимпиадой осуществляет структурное подразделение Национального исследовательского университета «Высшая школа экономики» — Центр Национальной технологической олимпиады. Организационный комитет по подготовке и проведению Национальной технологической олимпиады возглавляют первый заместитель Руководителя Администрации Президента Российской Федерации С. В. Кириенко и заместитель Председателя Правительства Российской Федерации Д. Н. Чернышенко.

Национальная технологическая олимпиада — это командная инженерная Олимпиада, позволяющая школьникам работать в самых передовых инженерных направлениях. Она базируется на опыте Олимпиады Кружкового движения НТИ и проводится с 2015 года, а с 2016 года входит в перечень Российского совета олимпиад школьников и дает победителям и призерам льготы при поступлении в университеты.

Всего заявки на участие в десятом юбилейном сезоне (2024–25 гг.) самых масштабных в России командных инженерных соревнованиях подали более 140 тысяч школьников. Общий охват олимпиады с 2015 года превысил 880 тысяч участников.

НТО способствует формированию профессиональной траектории школьников, увлеченных научно-техническим творчеством и помогает им:

- определить свой интерес в мире современных технологий;
- получить опыт решения комплексных инженерных задач;
- осознанно выбрать вуз для продолжения обучения и поступить в него на льготных условиях.

Кроме того, НТО позволяет каждому участнику познакомиться с перспективными направлениями технологического развития, ведущими экспертами и найти единомышленников.

Ценности НТО

Национальная технологическая олимпиада — командные инженерные соревнования для школьников и студентов. Олимпиада создает уникальное пространство, основанное на общих ценностях и смыслах, которыми делятся все участники процесса: школьники, студенты, организаторы, наставники и эксперты. В основе Олимпиады лежит представление о современном технологическом образовании как новом укладе жизни в быстро меняющемся мире. Эта модель предполагает:

- доступность качественного обучения для всех, кто стремится к знаниям;
- возможность непрерывного развития;
- совместное формирование среды, где гуманитарные знания и новые технологии взаимно усиливают друг друга.

Это — образ общества будущего, в котором участники Олимпиады оказываются уже сегодня.

Решать прикладные задачи, нацеленные на умножение общественного блага

В заданиях Олимпиады используются актуальные вызовы науки и технологий, адаптированные под уровень школьников. Они имеют прикладной характер и отражают реальные потребности общества, а системное и профессиональное решение подобных задач способствует развитию общего блага. Олимпиада предоставляет возможность попробовать себя в этом направлении уже сегодня и найти единомышленников.

Создавать, а не только потреблять

Стремление к созданию нового ценится выше потребления готового, а ориентация на общественную пользу — выше личной выгоды. Это не исключает заботу о собственных интересах, но подчеркивает: творчество приносит больше удовлетворения, чем пассивное потребление. Олимпиада — совместный труд организаторов, партнеров и участников, в котором важнее стремление решать общие задачи, чем критика чужих усилий.

Работать в команде

Командная работа рассматривается не только как эффективный способ достижения целей, но и как основа для формирования сообщества, объединенного общими ценностями. Команда помогает раскрыть индивидуальность каждого, при этом сохраняя уважение к другим. Такие горизонтальные связи необходимы для реализации амбициозных технологических проектов. Олимпиада способствует формированию подобного сообщества и приглашает к его созданию всех заинтересованных.

Осваивать и ответственно развивать новые технологии

Сообщество Национальной технологической олимпиады — часть Кружкового движения НТИ, объединенные интересом к современным технологиям, стремлением к их пониманию и созданию нового. Возможности технологий постоянно расширяются, однако развитие должно сопровождаться ответственностью. Этика инженера и ученого предполагает осознание последствий своих решений. Главное правило — создавая новое, не навредить.

Играть честно и пробовать себя

Ценится честная победа, достигнутая в рамках установленных правил. Это предполагает отказ от списывания, давления и манипуляций. Честная игра означает уважение к себе, команде и соперникам. Олимпиада поддерживается как безопасное пространство, где каждый может пробовать новое, не опасаясь ошибок, и постепенно становиться сильнее и увереннее в себе.

Быть человеком

Соревнования — это сложный и эмоционально насыщенный процесс, в котором особенно важны порядочность, вежливость и чуткость. Эмпатия, уважение и забота делают участие полезным и комфортным. Высоко ценится бережное отношение к людям и их труду, отказ от токсичной критики и готовность нести ответственность за слова и поступки. Участие в общем деле помогает не только окружающим, но и самому человеку.

Организационная структура НТО

НТО — межпредметная олимпиада. Спектр соревновательных направлений (профилей НТО) сформирован на основе актуального технологического пакета и связан с решением современных проблем в различных технологических отраслях. С полным перечнем направлений (профилей) можно ознакомиться на сайте НТО: https://ntcontest.ru/tracks/nto-school/.

Соревнования в рамках НТО проводятся по четырем трекам:

1. НТО Junior для школьников (5–7 классы).
2. НТО школьников (8–11 классы).
3. НТО студентов.
4. Конкурс цифровых портфолио «Талант НТО».

В 2024/25 учебном году 21 профиль НТО включен в Перечень олимпиад школьников, ежегодно утверждаемый Приказом Министерства науки и высшего образования Российской Федерации, а также в Перечень олимпиад и иных интеллектуальных и (или) творческих конкурсов, утверждаемый приказом Министерства просвещения Российской Федерации. Это дает право победителям и призерам профилей НТО поступать в вузы страны без вступительных испытаний (БВИ), получить 100 баллов ЕГЭ или дополнительные 10 баллов за индивидуальные достижения. Преимущества при поступлении победителям и призерам НТО предлагают более 100 российских вузов.

НТО для школьников 8–11 классов проводится в три этапа:

- Первый отборочный этап — заочный индивидуальный. Участникам предлагаются предметный тур, состоящий из задач по двум предметам, связанным с выбранным профилем, а также инженерный тур, задания которого погружают участников в тематику профиля; образовательный модуль формирует теоретические знания и представления.

- Второй отборочный этап — заочный командный. На этом этапе участники выполняют как индивидуальные задания на проверку компетенций, так и командные задачи, соответствующие выбранному профилю.
- Заключительный этап — очный командный. В течение 5–6 дней команды участников со всей страны, успешно прошедшие оба отборочных этапа, соревнуются в решении комплексных прикладных инженерных задач.

Профили НТО 2024/25 учебного года и соответствующий уровень РСОШ

Профили II уровня РСОШ:

- Автоматизация бизнес-процессов.
- Автономные транспортные системы.
- Беспилотные авиационные системы.
- Водные робототехнические системы.
- Инженерные биологические системы.
- Наносистемы и наноинженерия.
- Нейротехнологии и когнитивные науки.
- Технологии беспроводной связи.
- Цифровые технологии в архитектуре.
- Ядерные технологии.

Профили III уровня РСОШ:

- Анализ космических снимков и геопространственных данных.
- Аэрокосмические системы.
- Большие данные и машинное обучение.
- Геномное редактирование.
- Интеллектуальные робототехнические системы.
- Интеллектуальные энергетические системы.
- Информационная безопасность.
- Искусственный интеллект.
- Летающая робототехника.
- Спутниковые системы.
- Кластер «Виртуальные миры»:
    - ◇ Разработка компьютерных игр.
    - ◇ Технологии виртуальной реальности.
    - ◇ Технологии дополненной реальности.

Профили без уровня РСОШ:

- Инфохимия.
- Квантовый инжиниринг.
- Новые материалы.
- Программная инженерия в финансовых технологиях.

- Современная пищевая инженерия.
- Умный город.
- Урбанистика.
- Цифровые сенсорные системы.
- Разработка мобильных приложений.

Обратите внимание на то, что в олимпиаде 2025/26 учебного года список профилей, в т. ч. входящих в РСОШ, и уровни РСОШ могут поменяться.

Участие в НТО старшеклассников может принять любой школьник, обучающийся в 8–11 классе. Чаще всего Олимпиада привлекает:

- учащихся технологических кружков, интересующихся инженерными и робототехническими соревнованиями;
- школьников, увлеченных олимпиадами и предпочитающих межпредметный подход;
- энтузиастов передовых технологий;
- активных участников хакатонов, проектных конкурсов и профильных школ;
- будущих предпринимателей, ищущих команду для реализации стартап-идей;
- любознательных школьников, стремящихся выйти за рамки школьной программы.

Познакомить школьников с НТО и ее направлениями, а также мотивировать их на участие в Олимпиаде можно с помощью специальных мероприятий — Урока НТО и Дней НТО. Методические рекомендации для педагогов по проведению Урока НТО и организации Дня НТО в образовательной организации размещены на сайте: https://nti-lesson.ru. Здесь можно подобрать и скачать готовые сценарии занятий и подборки материалов по различным направлениям Олимпиады.

Участвуя в НТО, школьники получают возможность работать с практико-ориентированными задачами в области прорывных технологий, собирать команды единомышленников, погружаться в профессиональное сообщество, а также заработать льготы для поступления в вузы.

По всей стране работают площадки подготовки к НТО, которые помогают привлекать участников и проводят мероприятия по подготовке к этапам Олимпиады. Такие площадки могут быть открыты на базе:

- школ и учреждений дополнительного образования;
- частных кружков по программированию, робототехнике и другим технологическим направлениям;
- вузов;
- технопарков и других образовательных и научно-технических организаций.

Любое образовательное учреждение, ученики которого участвуют в НТО или НТО Junior, может стать площадкой подготовки к Олимпиаде и присоединиться к Кружковому движению НТИ. Подробные инструкции о том, как стать площадкой подготовки, размещены на сайте: https://ntcontest.ru. Условия регистрации и требования к ним актуализируются с развитием Олимпиады, а обновленная информация публикуется перед началом каждого нового цикла.

Наставники НТО

В Национальной технологической олимпиаде большое внимание уделяется работе с **наставниками** — людьми, сопровождающими участников на всех этапах подготовки и участия в Олимпиаде. Наставник оказывает поддержку как в решении организационных вопросов, так и в развитии технических и социальных навыков школьников, включая умение работать в команде.

Наставником НТО может стать любой взрослый, готовый помогать школьникам развиваться и готовиться к участию в инженерных соревнованиях. Это может быть:

- учитель школы или преподаватель вуза;
- педагог дополнительного образования;
- руководитель кружка;
- родитель школьника;
- специалист из технологической области или представитель бизнеса.

Даже если наставник сам не обладает достаточными знаниями в определенной области, он может привлекать к подготовке коллег и экспертов, а также оказывать поддержку и организовывать процесс обучения для самостоятельных учеников. Сегодня сообщество наставников НТО насчитывает более **7 000 человек** по всей стране.

Главная цель наставника — **организовать системную подготовку к Олимпиаде в течение всего учебного года**, поддерживать интерес и мотивацию участников, а также помочь им справляться с возникающими трудностями. Также наставник фиксирует цели команды и каждого участника, чтобы в дальнейшем можно было проанализировать развитие профессиональных и личных компетенций.

Основные направления работы наставника

Организационные задачи:

- Информирование и мотивация: наставник рассказывает учащимся об НТО, ее этапах и преимуществах, помогает с выбором подходящего профиля, ориентируясь на интересы и способности школьников.
- Составление программы подготовки: формируется расписание и план занятий, организуется работа по освоению необходимых знаний и навыков.
- Контроль сроков: наставник следит за календарем Олимпиады и напоминает участникам о сроках решения заданий отборочных этапов.

Содержательная подготовка:

- Оценка компетенций участников: наставник помогает определить сильные и слабые стороны учеников и подбирает задания и материалы для устранения пробелов.
- Подготовка к отборочным этапам: помощь в изучении рекомендованных материалов, заданий прошлых лет, онлайн-курсы по профилям.
- Подготовка к заключительному этапу: разбираются задачи заключительных этапов прошлых лет, отслеживаются подготовительные мероприятия (очные и дистанционные), в которых наставник рекомендует ученикам участвовать.

Развитие личных и командных навыков:

- Формирование команд: наставник помогает сформировать сбалансированные команды для второго отборочного и финального этапов, распределить роли, при необходимости ищет участников из других регионов и организует онлайн-коммуникацию.
- Анализ прогресса и опыта: после каждого этапа проводится совместная рефлексия, обсуждаются успехи и трудности, выявляются зоны роста и направления для дальнейшего развития.
- Поддержка и мотивация: наставник поддерживает интерес и энтузиазм участников (особенно в случае неудачных результатов), помогает справиться с разочарованием и сохранить настрой на дальнейшее участие.
- Построение индивидуальной образовательной траектории: наставник помогает школьникам осознанно планировать дальнейшее обучение: выбирать курсы, участвовать в конкурсах, определяться с вузами и направлениями подготовки.

Поддержка наставников НТО

Работе наставников посвящен отдельный раздел на сайте НТО: https://ntcontest.ru/mentors/.

Для систематизации знаний и подходов к работе наставников в рамках инженерных соревнований разработан курс «Дао начинающего наставника: как сопровождать инженерные команды»: https://stepik.org/course/124633/. Курс формирует общие представления об их работе в области подготовки участников к инженерным соревнованиям.

Для совершенствования профессиональных компетенций по направлениям профилей создан курс «Дао начинающего наставника: как развивать технологические компетенции»: https://stepik.org/course/186928/.

Для организации занятий с учениками педагогам предлагаются образовательные программы, разработанные на основе многолетнего опыта организации подготовки к НТО. В настоящий момент они представлены по передовым технологическим направлениям:

- компьютерное зрение;
- геномное редактирование;
- водная, летающая и интеллектуальная робототехника;
- машинное обучение и искусственный интеллект;
- нейротехнологии;
- беспроводная связь, дополненная реальность.

Программы доступны на сайте: https://ntcontest.ru/mentors/education-programs/.

Регистрируясь на платформе НТО, наставники получают доступ к личному кабинету, в котором отображается расписание отборочных соревнований и мероприятий по подготовке, требования к знаниям и компетенциям при решении задач отборочных этапов.

Сообщество наставников НТО существует и развивается. Ежегодно Кружковое движение НТИ проводит Всероссийский конкурс технологических кружков: https://konkurs.kruzhok.org/. Принять участие в конкурсе может каждый наставник.

В 2022 году было выпущено пособие «Технологическая подготовка инженерных команд. Методические рекомендации для наставников». Методические рекомендации предназначены для учителей технологий, а также наставников и педагогов кружков и центров дополнительного образования. Рекомендации направлены на помощь в процессе преподавания технологий в школе или в кружке. Пособие построено на примерах из реального опыта работы со школьниками, состоит из теоретических положений, посвященных популярным взглядам в педагогике на тему подготовки инженерных команд к соревнованиям. Электронное издание доступно по ссылке: https://journal.kruzhok.org/tpost/pggs3bp7y1-tehnologicheskaya-podgotovka-inzhenernih.

В нем рассмотрены особенности подготовки к пяти направлениям:

- Большие данные.
- Машинное обучение.
- Искусственный интеллект.
- Спутниковые системы.
- Летающая робототехника.

Для наставников НТО разработана и постоянно пополняется страница с материалами для профессионального развития: https://nto-forever.notion.site/c9b9cbd21542479b97a3fa562d15e32a.

1.2. Ядерные технологии

Профиль Ядерные технологии нацелен на задачи, связанные с развитием цифровизации в атомной энергетике и промышленности. Участникам предоставляется возможность построить цифровую модель и систему управления, которая функционирует в режиме реального времени в среде разработки, применяющейся в атомной отрасли для построения реальных систем управления.

Первый отборочный этап состоит из предметного и инженерного туров и включает в себя решение задач по двум школьным предметам: информатика и физика, а также проверку теоретических знаний по основам ядерных технологий и их применении в современной индустрии, особенно в ядерной энергетике.

Прочные знания по этим дисциплинам требуются для решения задач второго отборочного и заключительного этапов и, в частности, для понимания физических процессов в ядерной физике и процессов управления теми объектами, на которых применяются ядерные технологии.

На **втором отборочном этапе** участники знакомятся с легендой задачи заключительного этапа, а именно, построение математической модели объекта управления, выполняя задания по темам:

- физика;
- численное решение дифференциальных уравнений;
- программирование на языке С.

Используя методические материалы, участникам следует:

- подготовить систему дифференциальных уравнений, описывающих динамику объекта управления;
- построить разностную схему для системы дифференциальных уравнений по методу Эйлера;
- реализовать полученный на основе разностной схемы алгоритм расчета переходного процесса в виде программы на языке С;
- рассчитать значения динамических параметров предлагаемого объекта управления с помощью полученной программы.

На **заключительном этапе** все полученные на предыдущих этапах знания используются при работе с цифровой моделью объекта управления. Для реализованной модели выбирается закон управления, который поддерживает заданные технологические параметры в безаварийных диапазонах для всей реакторной установки.

Помимо этого предлагается сконструировать и реализовать человеко-машинный интерфейс, способный оперативно и безошибочно управлять реализованной моделью объекта и проводить эксперименты по исследованию и улучшению характеристик объекта управления.

Компетенции, приобретенные участниками в процессе выполнения индивидуальных и командных заданий, помогают им составить представление об основных специальностях ядерной отрасли, связанной с автоматизацией ядерных энергетических установок и построением киберфизических систем управления.

Финалисты Олимпиады могут проследить взаимосвязь между базовыми школьными дисциплинами — математикой, физикой и информатикой — и ядерными технологиями. Предложенные задания дают возможность школьникам понять, какие направления науки и техники важны для получения будущей специальности, связанной с автоматизацией процессов управления.

Профиль открывает для абитуриентов двери профильных вузов по таким направлениям, как «Ядерные физика и технологии», «Ядерная энергетика и теплофизика», «Приборостроение», «Автоматика и электроника физических установок», «Мехатроника и робототехника».

2. Первый отборочный этап

2.1. Работа наставника НТО на этапе

Педагог-наставник играет важную роль в подготовке участника к первому отборочному этапу Национальной технологической олимпиады. На этом этапе школьникам предстоит справиться как с предметными задачами, соответствующими профилю, так и с заданиями инженерного тура, погружающими в выбранную технологическую область.

Наставник может организовать подготовку участника, используя разнообразные форматы и ресурсы:

- Разбор заданий прошлых лет. Совместный анализ задач отборочного этапа предыдущих лет позволяет понять структуру, уровень сложности и типичные подходы к решению. Это формирует у школьника устойчивые стратегии работы с олимпиадными заданиями.
- Мини-соревнования. Проведение тренировочных турниров с заданиями предметных олимпиад муниципального уровня помогает развить соревновательный навык, тренирует скорость и уверенность при решении задач в ограниченное время.
- Углубленные занятия. Наставник может выстроить образовательную траекторию, опираясь на рекомендации разработчиков профиля, и провести занятия по ключевым темам. Это особенно важно для системного понимания предметной области.
- Использование онлайн-курсов. Для самостоятельной подготовки и проверки знаний участник может использовать предметные курсы НТО, размещенные на платформах Степик и Яндекс Контест. Наставник может также организовать занятия с использованием этих материалов в рамках групповой или индивидуальной подготовки.
- Привлечение внешних экспертов. Если у наставника нет достаточной экспертизы в какой-либо предметной области, он может пригласить других педагогов или специалистов для проведения тематических занятий.
- Поддержка в инженерном туре. Инженерный тур включает теоретические материалы и задания, помогающие глубже погрузиться в тематику профиля. Наставник может сопровождать изучение курса, помогать в разборе теоретических вопросов и тренировать участника на практических задачах.

Таким образом, наставник не только помогает систематизировать подготовку, но и мотивирует участника, создавая для него комфортную и продуктивную образовательную среду.

2.2. Предметный тур. Информатика

2.2.1. Первая волна. Задачи 8–11 класса

Задачи первой волны предметного тура по информатике открыты для решения. Соревнование доступно на платформе Яндекс.Контест: https://contest.yandex.ru/contest/63452/enter/.

Задача 2.2.1.1. Ускорение ускорения (10 баллов)

Имя входного файла: стандартный ввод или `input.txt`.

Имя выходного файла: стандартный вывод или `output.txt`.

Ограничение по времени выполнения программы: 1 с.

Ограничение по памяти: 64 Мбайт.

Условие

Рассмотрим модель движения тела. Будем фиксировать такие параметры, как координата, скорость, ускорение и ускорение ускорения (рывок). Если некоторый параметр равен a и имеет скорость изменения v, то в следующий момент времени этот параметр будет равен $a + v$.

Например, если тело имело координату, равную 10, скорость, равную 20, ускорение, равное 30 и ускорение ускорения, равное 40, то в следующий момент оно будет иметь координату 30, скорость 50 и ускорение 70. Ускорение ускорения будем считать в этой задаче постоянной величиной.

Задача довольно проста: тело в начальный момент времени 0 находится в точке с координатой 0, скоростью 0 и ускорением 0. На это тело действует постоянное ускорение ускорения, равное 6. Требуется определить, в точке с какой координатой окажется это тело в момент времени t.

Формат входных данных

В единственной строке находится одно число t, где $0 \leqslant t \leqslant 10^6$.

Формат выходных данных

Вывести одно число — координату, в которой окажется тело в момент времени t.

Примеры

Пример №1

Стандартный ввод
`6`

Стандартный вывод
`120`

Пример №2

Стандартный ввод
`2`

Стандартный вывод
`0`

Пример №3

Стандартный ввод
`1000000`

Стандартный вывод
`999997000002000000`

Решение

Ниже представлено решение на языке C++.

C++

```cpp
#include<bits/stdc++.h>
#define int long long
using namespace std;
signed main(){
    int t;
    cin >> t;
    cout << ((t * (t - 1)) * (t - 2)) << endl;
}
```

Задача 2.2.1.2. Двойное остекление (15 баллов)

Имя входного файла: стандартный ввод или `input.txt`.

Имя выходного файла: стандартный вывод или `output.txt`.

Ограничение по времени выполнения программы: 1 с.

Ограничение по памяти: 64 Мбайт.

Условие

У деда Василия есть два прямоугольных куска стекла. Один из них имеет размеры $a \times b$, другой — $c \times d$. Дед собирается из этих кусков сделать окно с двойным остеклением. Он хочет, чтобы окно было обязательно квадратным и как можно большим по размеру. Дед должен вырезать из имеющихся у него прямоугольников два одинаковых квадрата максимально возможного размера. Нужно написать программу, которая по заданным a, b, c, d найдет максимальные размеры квадратного окна. Имейте ввиду, что оба квадрата могут быть вырезаны и из одного прямоугольного куска стекла.

Формат входных данных

На вход подаются две строки. В первой строке находятся размеры первого прямоугольника a, b через пробел, во второй — размеры второго прямоугольника c, d через пробел, где $1 \leqslant a, b, c, d \leqslant 10^9$.

Формат выходных данных

Вывести одно число — максимальную сторону квадратного двойного окна, которое можно вырезать из заданных на входе прямоугольных кусков стекла. Ответ может быть нецелым, требуется вывести его с точностью 1 знак после десятичной точки.

Примеры

Пример №1

Стандартный ввод
5 10 9 6

Стандартный вывод
5

Пример №2

Стандартный ввод
4 10 9 6

Стандартный вывод
4.5

Комментарий

Второй пример показывает, что иногда лучше вырезать оба квадрата из одного и того же куска стекла.

Решение

Ниже представлено решение на языке C++.

C++

```cpp
#include<bits/stdc++.h>
#define int long long
using namespace std;
signed main(){
    double a, b, c, d;
    cin >> a >> b >> c >> d;
    double a0 = min({a, b, c, d});
    double a1 = min(max(a, b) / 2.0, min(a, b));
    double a2 = min(max(c, d) / 2.0, min(c, d));
    double ans =  max({a0, a1, a2});
    if( (int)ans == ans ){
        int ians = ans;
        cout << ians << endl;
        return 0;
    }
    cout.precision(1);
    cout << fixed<< ans << endl;
}
```

Задача 2.2.1.3. О золотой рыбке и... досках (20 баллов)

Имя входного файла: стандартный ввод или `input.txt`.

Имя выходного файла: стандартный вывод или `output.txt`.

Ограничение по времени выполнения программы: 1 с.

Ограничение по памяти: 64 Мбайт.

Условие

После событий известной сказки А. С. Пушкина старик решил принципиально не пользоваться услугами золотой рыбки. Поэтому для того чтобы изготовить новое корыто, он честно заготовил n одинаковых досок.

Но гостивший в это время у старика со старухой внук решил, что ему нужно научиться пилить. И, не сказав ничего своему деду, внук быстро распилил каждую из досок на две части. В итоге у старика оказались $2n$ кусков досок. Самое интересное, что все эти куски оказались разными по длине, но имели целочисленные размеры. К сожалению, старик забыл, какова была исходная длина целых досок.

Формат входных данных

В первой строке задается целое число n — исходное количество целых досок, где $1 \leqslant n \leqslant 10^5$.

Во второй строке заданы $2n$ целых чисел d_i — длины всех кусков, которые получились после «тренировки» внука, где $1 \leqslant d_i \leqslant 10^9$. Гарантируется, что эти числа попарно различны, и их можно разбить на пары одинаковых по сумме чисел.

Все эти части досок пронумерованы от 1 до $2n$ в том порядке, в котором они заданы на входе.

Формат выходных данных

В первую строку вывести одно число — исходную длину целых досок.

В следующих n строках вывести пары номеров кусков досок, которые составляют по длине целые доски. Номера выводить через один пробел, внутри пары сначала должен идти меньший номер, затем больший. Пары должны быть выведены в порядке возрастания первых номеров в парах.

Примеры

Пример №1

Стандартный ввод
3 4 8 2 3 6 7

Стандартный вывод
10 1 5 2 3 4 6

Комментарий

Отсортируем куски и далее будем брать один из начала и второй к нему из конца.

Решение

Ниже представлено решение на языке C++.

C++

```cpp
#include<bits/stdc++.h>
#define int long long
using namespace std;
signed main(){
    int n;
    cin >> n;
    vector<pair<int, int> > v(2 * n);
     for(int i = 0; i < 2 * n; i++){
        int d;
        cin >> d;
        v[i] = {d, i + 1};
    }
    sort(v.begin(), v.end());
    vector<pair<int, int> > ans(n);
    for(int i = 0; i < n; i++){
```

```
        ans[i] = {v[i].second, v[2 * n - i - 1].second};
        if(ans[i].first > ans[i].second){
            swap(ans[i].first, ans[i].second);
        }
    }
    sort(ans.begin(), ans.end());
    cout << v[0].first + v.back().first<< endl;
    for(int i = 0; i < n; i++){
        cout << ans[i].first<<' '<< ans[i].second<< endl;
    }
}
```

Задача 2.2.1.4. Бонусы и экономия (25 баллов)

Имя входного файла: стандартный ввод или `input.txt`.

Имя выходного файла: стандартный вывод или `output.txt`.

Ограничение по времени выполнения программы: 1 с.

Ограничение по памяти: 64 Мбайт.

Условие

Технология производства некоторой металлической детали предполагает вытачивание ее из металлической заготовки. При этом образуются стружки, которые не стоит выкидывать. Ведь из a комплектов стружек (оставшихся после обработки a заготовок) можно бесплатно выплавить еще одну заготовку, которую снова можно использовать для выточки детали и создания еще одного комплекта стружек.

Заготовки можно купить на оптовом складе, при этом в целях привлечения клиентов, проводится акция «купи b заготовок, тогда еще одну получишь бесплатно».

Требуется изготовить c деталей. Нужно определить минимальное число заготовок, которые нужно купить за деньги, чтобы с учетом бонусных заготовок и экономии на стружках можно было изготовить требуемое число деталей.

Формат входных данных

В одной строке через пробел заданы три целых числа a, b, и c такие, что $2 \leqslant a \leqslant 10^{18}$, $1 \leqslant b, c \leqslant 10^{18}$.

Формат выходных данных

Вывести одно целое число — минимальное количество заготовок, которые нужно купить, чтобы с учетом всех бонусов и экономии выточить c конечных деталей.

Примеры

Пример №1

Стандартный ввод
4 5 41

Стандартный вывод
26

Примечания

В примере из условия нужно закупить 26 заготовок. Тогда за каждые пять купленных заготовок будет предоставлена одна бесплатная, итого по акции добавится еще пять заготовок, то есть получится 31 заготовка. Далее из 31 заготовки выточится 31 деталь, останется 31 комплект стружек. Из каждых четырех комплектов выплавится дополнительная заготовка, получится семь заготовок и три комплекта стружек. Из семи заготовок выточится семь деталей и останется семь комплектов стружек, три комплекта стружек осталось с первого шага, итого 10 комплектов стружек. Из них выплавится еще две заготовки, дающие две детали и два комплекта стружек. Собрав эти два комплекта с двумя, оставшимися от 10, получим еще одну заготовку, из которой выточится еще одна деталь. Останется один комплект стружек, который уже никак не получится использовать. Итого будет произведена $31 + 7 + 2 + 1 = 41$ деталь.

Комментарий

Методом бинарного поиска можно подобрать минимальное необходимое количество исходных заготовок.

Решение

Ниже представлено решение на языке C++.

C++

```cpp
#include<bits/stdc++.h>
#define int long long
using namespace std;
int f1(int M, int a){
    int res = 0, z = 0;
    while(1){
        if(M == 0 && z < a){
            return res;
        }
        res += M;
        M = M + z;
        z = M % a;
        M = M / a;
    }
}
```

```
int f2(int M, int b){
    return M + M / b;
}
signed main(){
    int a, b, c;
    cin >> a >> b >> c;
    int L = 0, R = 1;
    while(f1(R, a) <= c){
        R *= 2;
    }
    while(R - L > 1){
        int M = (R + L) / 2;
        if(f1(M, a) < c){
            L = M;
        }
        else{
            R = M;
        }
    }
    int z = R;
    L = 0, R = 1;
    while(f2(R, b) <= z){
        R *= 2;
    }
    while(R - L > 1){
        int M = (R + L) / 2;
        if(f2(M, b) < z){
            L = M;
        }
        else{
            R = M;
        }
    }
    cout << R << endl;
}
```

Задача 2.2.1.5. Сон таксиста (30 баллов)

Имя входного файла: стандартный ввод или `input.txt`.

Имя выходного файла: стандартный вывод или `output.txt`.

Ограничение по времени выполнения программы: 1 с.

Ограничение по памяти: 64 Мбайт.

Условие

Одному таксисту приснился красочный сон. Во сне он живет и работает в некотором городе, где абсолютно все улицы с односторонним движением. Эти улицы устроены так, что невозможно проехать с какого-либо перекрестка так, чтобы вернуться обратно на этот же перекресток, то есть в дорожной сети города нет циклов.

Таким образом, если с перекрестка A можно попасть по направлению движения улиц на перекресток B, то люди вызывают такси, иначе их везет специальный муниципальный подземный транспорт бесплатно.

В связи с такими странными правилами, таксистам в этом городе разрешено законом везти пассажира по любому маршруту, не нарушающему направления движения. Все в этом городе привыкли к такой ситуации и абсолютно спокойно относятся к тому, что таксисты везут их самым длинным путем. Разумеется, заработок таксиста за одну поездку прямо пропорционален ее длине. Для упрощения будем считать, что стоимость 1 км поездки составляет ровно 1 руб.

Схема дорог города задана. Перекрестки города пронумерованы числами от 1 до n. Таксист в своем сне находится на перекрестке номер S. Напишите программу, которая подскажет ему, сколько он максимально сможет заработать, когда ему придет заказ от клиента. Так как он не знает, куда попросит его везти клиент, нужно для каждого перекрестка от 1 до n указать максимальную стоимость поездки до этого перекрестка из пункта S на такси. Если по правилам на такси добраться из пункта S до какого-то перекрестка нельзя, вывести -1.

Формат входных данных

Дорожная сеть задана следующим образом: в первой строке находятся два числа через пробел n и m — число перекрестков и число улиц в городе, где $2 \leqslant n, m \leqslant 2 \cdot 10^5$.

В следующих m строках задана очередная односторонняя улица в виде трех чисел A, B, d через пробел, где A — начало улицы, B — конец улицы и d — ее длина. $1 \leqslant A, B \leqslant n$, $1 \leqslant d \leqslant 10^9$. Гарантируется, что в этой дорожной сети нет циклов. Некоторые пары перекрестков могут быть соединены двумя и более односторонними улицами. Дорожная сеть может быть неплоской за счет мостов и тоннелей.

В последней строке ввода содержится номер стартового перекрестка S, $1 \leqslant S \leqslant n$.

Формат выходных данных

Вывести n чисел в одну строку через пробел. i-е число обозначает длину самого длинного пути с перекрестка номер S до перекрестка номер i. Если до перекрестка номер i от S нельзя доехать, не нарушая правила движения, вывести -1.

Примеры

Пример №1

Стандартный ввод
10 20 9 10 15 9 8 3 8 10 7 7 8 4 7 10 10 5 8 2 5 9 10

Стандартный ввод
5 6 5 7 6 5 4 6 8 3 6 4 3 4 6 5 3 2 2 5 2 2 3 3 3 1 5 1 4 2 2 1 7 4 7 4 6 8 1 5

Стандартный вывод
7 -1 2 9 0 18 13 19 10 26

Комментарий

Задача решается методом динамического программирования на ориентированном ациклическом графе.

Решение

Ниже представлено решение на языке C++.

C++

```cpp
#include<bits/stdc++.h>
#define int long long
using namespace std;
int n, m;
vector<vector<pair<int, int> > > G;
vector<int> order, used;
void dfs(int a){
    used[a] = 1;
    for(auto to : G[a]){
        if(!used[to.first]){
            dfs(to.first);
        }
    }
    order.push_back(a);
}
signed main(){
    cin >> n >> m;
    G.resize(n + 1);
    used.resize(n + 1, 0);
    for(int i = 0; i < m; i++){
        int a, b, d;
        cin >> a >> b >> d;
        G[a].push_back({b, d});
    }
```

```
    int s;
    cin >> s;
    dfs(s);
    reverse(order.begin(), order.end());
    vector<int> dp(n + 1, -1);
    dp[s] = 0;
    for(auto el : order){
        for(auto to : G[el]){
            dp[to.first] = max(dp[to.first], dp[el] + to.second);
        }
    }
    for(int i = 1; i <= n; i++){
        cout << dp[i] <<' ';
    }
}
```

2.2.2. Вторая волна. Задачи 8–11 класса

Задачи второй волны предметного тура по информатике открыты для решения. Соревнование доступно на платформе Яндекс.Контест: https://contest.yandex.ru/contest/63454/enter/.

Задача 2.2.2.1. Игра на планшете (10 баллов)

Имя входного файла: стандартный ввод или input.txt.

Имя выходного файла: стандартный вывод или output.txt.

Ограничение по времени выполнения программы: 1 с.

Ограничение по памяти: 64 Мбайт.

Условие

Маленький Андрей изучает геометрические фигуры при помощи игры на планшете. У него есть прямоугольные треугольники четырех цветов и ориентаций: желтые, зеленые, красные и синие. Для каждой разновидности треугольников есть заданное количество экземпляров этих треугольников. Более точно: у Андрея есть a желтых, b зеленых, c красных и d синих треугольников. Помимо этого у него есть прямоугольная таблица $n \times m$.

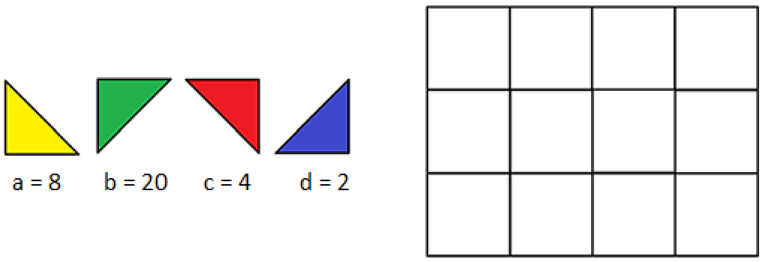

Рис. 2.2.1

Треугольники одного цвета имеют одну и ту же ориентацию, которую нельзя поменять. Андрей может только взять очередной треугольник и переместить его параллельным сдвигом в одну из ячеек этой прямоугольной таблицы. При этом в одну ячейку можно поместить либо вместе желтый и красный треугольники, либо вместе зеленый и синий, либо один любой треугольник из имеющихся.

Андрей хочет расположить в ячейках таблицы как можно больше треугольников из тех, что у него имеются. Нужно подсказать ему максимальное количество треугольников, которые получится разместить в таблице.

Формат входных данных

В первой строке содержатся четыре целых числа a, b, c и d через пробел — количество желтых, зеленых, красных и синих треугольников соответственно.

Во второй строке содержатся два целых числа n и m через пробел — размеры прямоугольной таблицы.

Все числа в пределах от 1 до 10^9.

Формат выходных данных

Вывести одно число — максимальное количество треугольников, которые можно при заданных условиях разместить в таблице.

Примеры

Пример №1

Стандартный ввод
`8 20 4 2` `3 4`

Стандартный вывод
`18`

Примечания

На рис. 2.2.2 представлен один из примеров размещения 18 треугольников из 34 заданных на входе.

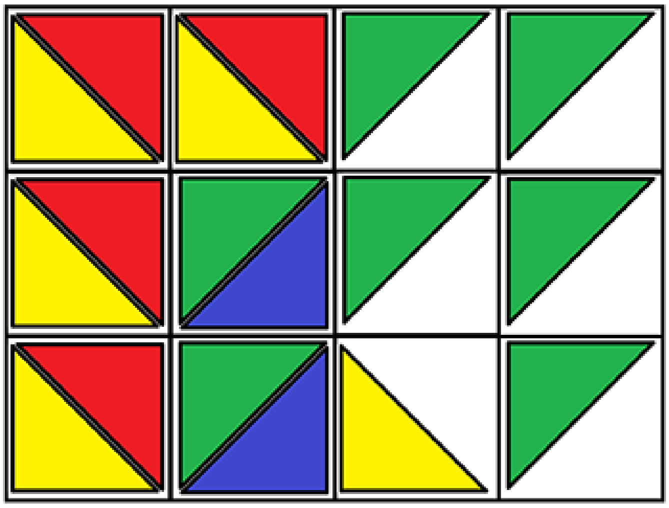

Рис. 2.2.2

Решение

Ниже представлено решение на языке C++.

C++

```
#include<bits/stdc++.h>
#define int long long
using namespace std;
signed main(){
    int a, b, c, d, n, m;
    cin >> a >> b >> c >> d >> n >> m;
    if(a > c){
        swap(a, c);
    }
    if(b > d){
        swap(b, d);
    }
    int f = a + b;
    int k = n * m;
    if(k <= f){
        cout << k * 2;
        return 0;
    }
    k -= f;
    c -= a;
    d -= b;
    cout << f * 2 + min(k, c + d) << endl;
}
```

Задача 2.2.2.2. Старая задача на новый лад (15 баллов)

Имя входного файла: стандартный ввод или `input.txt`.

Имя выходного файла: стандартный вывод или `output.txt`.

Ограничение по времени выполнения программы: 1 с.

Ограничение по памяти: 64 Мбайт.

Условие

Одна старая задача имеет следующий вид:

«Разбить число 45 на сумму четырех слагаемых так, что если к первому прибавить 2, из второго вычесть 2, третье умножить на 2, а четвертое разделить на 2, то получится одно и то же число».

Ответ к этой задаче — четыре числа 8, 12, 5 и 20. Можно убедиться, что в сумме они дают число 45, а если с каждым из них проделать соответствующую арифметическую операцию, то получится одно и то же число 10.

Необходимо решить чуть более общую задачу: даны числа n и k. Нужно представить число n в виде суммы четырех целых неотрицательных слагаемых $a + b + c + d$ таких, что $a + k = b - k = c \cdot k = d/k$. Гарантируется, что для заданных n и k такое разбиение существует.

Формат входных данных

В одной строке через пробел два числа n и k, где $1 \leqslant n \cdot k \leqslant 10^{18}$.

Формат выходных данных

Вывести через пробел в одну строку четыре целых неотрицательных числа a, b, c, d таких, что $a + b + c + d = n$ и $a + k = b - k = c \cdot k = d\,/\,k$.

Примеры

Пример №1

Стандартный ввод
`45 2`

Стандартный вывод
`8 12 5 20`

Пример №2

Стандартный ввод
`128 7`

Стандартный вывод
`7 21 2 98`

Решение

Ниже представлено решение на языке C++.

C++

```cpp
#include<bits/stdc++.h>
#define int long long
using namespace std;
signed main(){
    int n, k;
    cin >> n >> k;
    int x = (k * n) / (k * k + 2 * k + 1);
    cout << x - k <<' '<< x + k <<' '<< x / k <<' '<< x * k << endl;
}
```

Задача 2.2.2.3. Ладья и обязательная клетка (20 баллов)

Имя входного файла: стандартный ввод или `input.txt`.

Имя выходного файла: стандартный вывод или `output.txt`.

Ограничение по времени выполнения программы: 1 с.

Ограничение по памяти: 64 Мбайт.

Условие

Шахматная ладья находится в левом верхнем углу прямоугольного поля, разбитого на клетки размером $n \times m$. n обозначает число строк, m — число столбцов. Она хочет попасть в правую нижнюю клетку этого поля кратчайшим путем. Ладья может передвигаться либо вправо, либо вниз на любое количество клеток. Ладья обязана посетить заданную клетку с координатами (x, y), где x — номер строки этой клетки, а y — номер ее столбца.

Требуется найти количество способов построить путь ладьи из левого верхнего угла в правый нижний, которые проходят через обязательную клетку с заданными координатами.

Формат входных данных

В первой строке находятся два числа через пробел: n — число строк и m — число столбцов прямоугольного поля, $2 \leqslant n, m \leqslant 25$. Во второй строке через пробел находятся координаты (x, y) обязательной для посещения клетки, где $1 \leqslant x \leqslant n$, $1 \leqslant y \leqslant m$. Координаты x и y не совпадают с координатами левой верхней и правой нижней клеток.

Формат выходных данных

Вывести одно число — количество кратчайших путей ладьи из верхней левой в правую нижнюю клетку, проходящих через заданную клетку.

Примеры

Стандартный ввод
3 4 2 3

Стандартный вывод
6

Примечания

На рис. 2.2.3 представлены шесть путей, которыми ладья может пройти по полю размером 3×4, обязательно посещая по пути клетку $(2, 3)$.

Комментарий

Задачу можно решить как комбинаторными методами (произведение биномиальных коэффициентов), так и динамическим программированием.

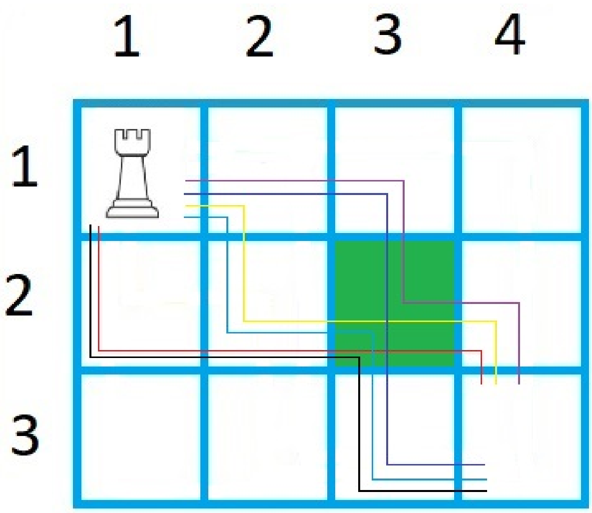

Рис. 2.2.3

Решение

Ниже представлено решение на языке C++.

C++

```
#include<bits/stdc++.h>
#define int long long
using namespace std;
signed main(){
    vector<vector<int> > bc(51, vector<int>(51, 0));
    bc[0][0] = 1;
    for(int i = 1; i <= 50; i++){
        for(int j = 0; j < 51; j++){
```

```
            bc[i][j] += bc[i - 1][j];
            if(j - 1 >= 0){
                bc[i][j] += bc[i - 1][j - 1];
            }
        }
    }
    int n, m, x, y;
    cin >> n >> m >> x >> y;
    int d1 = bc[x - 1 + y - 1][x - 1];
    int d2 = bc[n - x + m - y][n - x];
    int ans = d1 * d2;
    cout << ans << endl;
}
```

Задача 2.2.2.4. Танец с цифрами (25 баллов)

Имя входного файла: стандартный ввод или `input.txt`.

Имя выходного файла: стандартный вывод или `output.txt`.

Ограничение по времени выполнения программы: 1 с.

Ограничение по памяти: 64 Мбайт.

Условие

Десять танцоров репетируют на сцене новый танец. Каждый танцор одет в футболку, на которой написана одна из цифр от 1 до 9, цифры могут повторяться. Изначально они стоят в некотором порядке слева направо, и их цифры образуют некоторое десятизначное число A. Далее во время всего танца участники либо разбиваются на пять пар рядом стоящих танцоров и одновременно меняются местами внутри своих пар, либо самый левый танцор перемещается на самую правую позицию и становится самым правым танцором.

Сын постановщика танца от скуки на бумаге выписывает все получающиеся при каждом перемещении десятизначные числа. Так как танец длинный, то в итоге на бумаге окажутся все возможные числа, которые в принципе могут появится при этих условиях. Нужно найти разницу между самым большим и самым маленьким из этих чисел.

Формат входных данных

На вход подается одно десятизначное число A, обозначающее начальное расположение танцоров. В числе могут встречаться цифры от 1 до 9, некоторые из них могут повторяться.

Формат выходных данных

Вывести одно число, равное разности самого большого и самого маленького из чисел, которые могут быть получены во время танца.

Примеры

Пример №1

Стандартный ввод
1456531355

Стандартный вывод
5182160085

Примечания

Самое маленькое число, которое можно получить в примере, равно 1353155456, самое большое равно 6535315541.

Покажем, как получить эти числа из исходного числа 1456531355. Сначала получим самое большое следующим образом: две левых цифры, 1 и 4, переместим вправо, получим 5653135514, потом поменяем в парах цифры местами и получим самое большое — 6535315541. Далее опять поменяем порядок в парах и в числе 5653135514 переместим три левых цифры 5, 6 и 5 вправо, получим 3135514565 и здесь снова поменяем порядок в парах, получим самое маленькое — 1353155456. Таким образом, искомая разница равна 5182160085.

Решение

Ниже представлено решение на языке C++.

C++

```cpp
#include<bits/stdc++.h>
#define int long long
using namespace std;
signed main(){
    string s;
    cin >> s;
    string mx = s, mn = s;

    for(int i = 0; i < 5; i++){
        for(int j = 0; j < 10; j++){
            mx = max(mx, s);
            mn = min(s, mn);
            if(j < 9){
                s = s.substr(1) + s[0];
            }
        }
        for(int j = 0; j < 5; j++){
            swap(s[2 * j], s[2 * j + 1]);
        }
    }
    stringstream ssmn;
    ssmn << mn;
    int imn;
    ssmn >> imn;
    stringstream ssmx;
```

```
    ssmx << mx;
    int imx;
    ssmx >> imx;
    cout << imx - imn << endl;
}
```

Задача 2.2.2.5. Трудная сортировка (30 баллов)

Имя входного файла: стандартный ввод или `input.txt`.

Имя выходного файла: стандартный вывод или `output.txt`.

Ограничение по времени выполнения программы: 3 с.

Ограничение по памяти: 64 Мбайт.

Условие

Иннокентий работает в отделе сортировки перестановок, подотделе сортировки вставками. Его задача заключается в сортировке перестановок, предоставленных заказчиками. Перестановкой длины n называется такая последовательность чисел, в которой встречаются все числа от 1 до n без повторений в некотором порядке.

Перестановка считается отсортированной, если в ней все числа расположены по возрастанию, то есть она имеет вид $1, \ldots, n$.

Иннокентий начинает рабочий день с пустой последовательности чисел. За день он сортирует вставками перестановку длины n. В начале каждой операции вставки он получает очередное число a_i из перестановки заказчика, после чего обрабатывает его, вставляя в отсортированную последовательность из ранее полученных чисел. После каждого такого добавления последовательность уже обработанных чисел должна быть отсортирована по возрастанию.

Перед тем как вставить число a_i в последовательность, он может выбрать, с какого края последовательности начать вставку. Далее он устанавливает число a_i с этого края и последовательно меняет вставляемое число с рядом стоящим числом b_j до тех пор, пока число a_i не встанет на свое место. На каждую перестановку вставляемого числа a_i с числом b_j Иннокентий тратит b_j единиц энергии.

Дана перестановка длины n из чисел a_i в том порядке, в котором Иннокентий их будет обрабатывать. Подскажите ему, какое минимальное количество энергии ему потребуется потратить, чтобы отсортировать всю перестановку.

Формат входных данных

В первой строке находится одно целое число n — длина перестановки, где $1 \leqslant n \leqslant 2 \cdot 10^5$.

Во второй строке содержится n целых чисел a_i через пробел в том порядке, в котором они поступают на обработку Иннокентию. Гарантируется, что эти числа образуют перестановку длины n, то есть каждое число от 1 до n содержится в заданном наборе ровно один раз.

Формат выходных данных

Вывести одно число — минимальные суммарные энергозатраты Иннокентия для сортировки вставками заданной на входе перестановки.

Примеры

Пример №1

Стандартный ввод
9 2 9 1 5 6 4 3 8 7

Стандартный вывод
43

Примечания

Первым устанавливается число 2. Оно ни с чем не меняется местами, поэтому затрат нет.

Далее устанавливается число 9. Выбираем правый край и ставим его туда без потерь энергии.

Затем устанавливаем число 1. Выбираем левый край, ставим его туда и снова потерь нет.

Теперь нужно вставить число 5. Если его вставлять с правого края, придется менять местами с 9, а если с левого, то с 1 и 2, что суммарно явно лучше. Итого затраты на вставку 5 равны 3.

Число 6 снова лучше вставить слева, затраты на его вставку равны 8.

Число 4 вставим слева за 3.

Число 3 так же слева за 3.

А вот число 8 лучше вставить справа за 9.

И осталось число 7. Если вставлять слева, то затратим 21, а если справа, то всего 17.

Итого на сортировку заданной перестановки потратили: $0 + 0 + 0 + 3 + 8 + 3 + 3 + 9 + 17 = 43$.

Комментарий

Построим дерево отрезков на сумму, при обработке числа a будем находить, какая сумма на данный момент меньше: от 1 до $a-1$ или от $a+1$ до n. Прибавим ее к ответу и поместим в позицию a это число a.

Решение

Ниже представлено решение на языке C++.

C++

```cpp
#include<bits/stdc++.h>
#define int long long
using namespace std;
const int LG = 19;
int N = (1 << LG);
vector<int> tr(2 * N, 0);
void upd(int pos, int x){
    pos += N;
    tr[pos] = x;
    pos /= 2;
    while(pos){
        tr[pos] = {tr[2 * pos]+ tr[2 * pos + 1]};
        pos /= 2;
    }
}
int get(int l, int r){
    l += N;
    r += N;
    int res = 0;
    while(l <= r){
        if(l % 2 == 1){
            res += tr[l];
        }
        if(r % 2 == 0){
            res += tr[r];
        }
        l = (l + 1) / 2;
        r = (r - 1) / 2;
    }
    return res;
}
signed main(){
    int n, a;
    cin >> n;
    int ans = 0;
    for(int i = 0; i < n; i++){
        cin >> a;
        int sl = get(0, a - 1);
        int sr = get(a + 1, N - 1);
        ans += min(sl, sr);
        upd(a, a);
    }
    cout << ans << endl;
}
```

2.2.3. Третья волна. Задачи 8–11 класса

Задачи третьей волны предметного тура по информатике открыты для решения. Соревнование доступно на платформе Яндекс.Контест: https://contest.yandex.ru/contest/63456/enter/.

Задача 2.2.3.1. Туннель (10 баллов)

Имя входного файла: стандартный ввод или `input.txt`.

Имя выходного файла: стандартный вывод или `output.txt`.

Ограничение по времени выполнения программы: 1 с.

Ограничение по памяти: 64 Мбайт.

Условие

Рассмотрим классическую задачу прохождения группы с одним фонариком по туннелю. Есть четыре человека, и у них есть один фонарик. Нужно перевести всю группу на другой конец туннеля. По туннелю можно проходить только с фонариком и только либо вдвоем, либо в одиночку. По этой причине придется сделать пять рейсов по туннелю: три рейса туда и два рейса обратно. Туда идут двое, обратно — один, возвращая фонарик еще не прошедшей части группы. У каждого из четырех человек своя скорость передвижения по туннелю, но некоторые скорости могут совпадать. Двое идут со скоростью самого медленного в этой паре. Нужно найти минимальное время, за которое можно перевести группу по туннелю.

Здесь, в зависимости от скоростей персонажей, есть две стратегии. Проиллюстрируем их на примерах.

Пусть есть люди A, B, C, D. У A — время прохождения туннеля 1 мин, у B — 4 мин, у C — 5 мин, у D — 10 мин. Здесь работает наиболее очевидная стратегия: самый быстрый переводит текущего и возвращается с фонариком обратно за следующим. При этой стратегии нужно проходить так:

- A, B туда, затрачено 4 мин;
- A обратно, затрачена 1 мин;
- A, C туда, затрачено 5 мин;
- A обратно, затрачена 1 мин;
- A, D туда, затрачено 10 мин.

Общее время $4 + 1 + 5 + 1 + 10 = 21$ мин.

Но не всегда эта стратегия оптимальна. Уменьшим время прохождения туннеля персонажем B до 2 мин. По вышеопределенной стратегии будет 19 мин ($2 + 1 + 5 + 1 + 10 = 19$), но имеется более быстрое решение:

- A, B туда, затрачено 2 мин;
- A обратно, затрачена 1 мин;
- C, D туда, затрачено 10 мин;
- B обратно, затрачено 2 мин;
- A, B туда, затрачено 2 мин.

Общее время $2 + 1 + 10 + 2 + 2 = 17$ мин.

Заметим, что для предыдущего примера такая стратегия не работает: $4+1+10+4+4 = 23$ мин.

Если же персонаж B проходит туннель за 3 мин (а все остальные так же, как и в примерах), то независимо от стратегии будет затрачено 20 мин. В этом случае считаем, что работает первая стратегия.

Поразмыслив, станет понятно, от какого условия зависит выбор стратегии. Далее будем всегда считать, что A движется не медленнее B, B движется не медленнее C, C движется не медленнее D.

Дано время прохождения туннеля персонажами A, C, D. Нужно найти границу `border` для B такую, что если определить для B время прохождения строго меньшее, чем `border`, то выгодна вторая стратегия, иначе — первая.

Формат входных данных

В одной строке задано три целых чисел через пробел — время прохождения туннеля персонажами A, C, D. Времена даны по неубыванию. Все числа на входе в пределах от 1 до 100.

Формат выходных данных

Вывести одно число — границу `border` для B такую, что если определить время прохождения им туннеля строго меньше, чем `border`, нужно использовать вторую стратегию, иначе — первую. Ответ может быть нецелым, поэтому вывести его нужно с одним знаком после десятичной точки.

Примеры

Пример №1

Стандартный ввод
`1 5 10`

Стандартный вывод
`3`

Решение

Ниже представлено решение на языке C++.

C++

```cpp
#include<bits/stdc++.h>
#define int long long
using namespace std;
signed main(){
    int A, C, D;
    cin >> A >> C >> D;
    cout.precision(1);
    cout << fixed << (A + C) / 2.0 << endl;
}
```

Задача 2.2.3.2. Математический пазл (15 баллов)

Имя входного файла: стандартный ввод или `input.txt`.

Имя выходного файла: стандартный вывод или `output.txt`.

Ограничение по времени выполнения программы: 1 с.

Ограничение по памяти: 64 Мбайт.

Условие

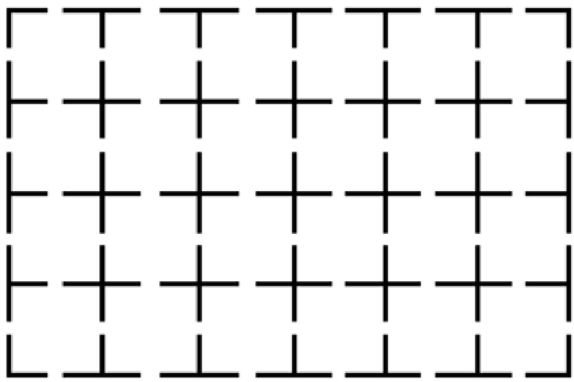

Рис. 2.2.4

Компания по производству пазлов решила освоить принципиально новый тип головоломок. Для этого берется прямоугольная решетка размера $n \times m$, каждый ее столбец и строка разрезаются посередине пополам. После этого образуются фигуры трех типов: четыре уголка, $2 \cdot (n+m-2)$ т-образных фигур и $(n-1) \cdot (m-1)$ крестиков.

Тому, кто решает головоломку, требуется сложить из этих фигур исходную прямоугольную решетку. При этом необходимо использовать абсолютно все имеющиеся в наличии фигуры.

Формат входных данных

В первой строке заданы через пробел два числа a — количество т-образных фигур и b — количество крестиков, которые находятся в одном из пазлов. При этом в наборе всегда есть еще четыре уголка. Известно, что этот комплект позволяет собрать прямоугольную решетку размера $n \times m$, где $1 \leqslant n,\ m \leqslant 10^9$.

Формат выходных данных

Требуется по числам a и b найти размеры исходной решетки n и m. Будем всегда считать, что $n \leqslant m$, то есть нужно вывести в одну строку через пробел два числа, первое из которых не превосходит второго, и вместе они задают размеры загаданной решетки.

Примеры

Пример №1

Стандартный ввод
16 15

Стандартный вывод
4 6

Пример №2

Стандартный ввод
0 0

Стандартный вывод
1 1

Комментарий

Задачу можно решить либо бинарным поиском, либо при помощи квадратного уравнения.

Решение

Ниже представлено решение на языке C++ при помощи бинпоиска.

C++

```cpp
#include<bits/stdc++.h>
#define int long long
using namespace std;
signed main(){
    int a, b;
    cin >> a >> b;
    int L = 0, R = a / 4 + 1;
    while(R - L > 1){
        int M = (R + L) / 2;
        int D = a / 2 - M;
        if(M * D <= b){
            L = M;
        }
        else{
            R = M;
        }
    }
    cout << L + 1 <<' '<< a / 2 - L + 1 << endl;
}
```

Задача 2.2.3.3. Восемь пирогов и одна свечка (20 баллов)

Имя входного файла: стандартный ввод или `input.txt`.

Имя выходного файла: стандартный вывод или `output.txt`.

Ограничение по времени выполнения программы: 1 с.

Ограничение по памяти: 64 Мбайт.

Условие

Мечта Карлсона наконец-то сбылась! Мама Малыша испекла восемь пирогов прямоугольной формы и в один из них воткнула свечку. После того как Карлсон съел семь пирогов, он решил-таки поделиться кусочком оставшегося восьмого пирога с Малышом. Но, будучи в хорошем настроении, он вынул из пирога свечу и предложил ему решить задачку.

«Так как я самый щедрый Карлсон в мире, то делить оставшийся пирог будешь ты. Но учти, ты должен разрезать пирог одним прямым разрезом так, чтобы линия прошла через один из углов и точку, где стояла свечка. После этого я выберу себе один из двух кусочков, а оставшийся, так и быть, достанется тебе».

Малыш не против этого замысла, однако считает, что разрезать пирог нужно как можно более справедливо, то есть так, чтобы разница между меньшим и большим кусками была как можно меньше. Подскажите Малышу, какой минимальной разницы между площадями кусков он сможет добиться.

Формат входных данных

В первой строке находятся два числа n и m через пробел — размеры прямоугольного пирога. Пирог размещен на координатной плоскости так, что его левый нижний угол находится в точке $(0, 0)$, а правый верхний — в точке (n, m), где $2 \leqslant n,\ m \leqslant 1\,000$.

Во второй строке находятся два числа x и y через пробел — координаты свечки, где $1 \leqslant x \leqslant n - 1$, $1 \leqslant y \leqslant m - 1$, то есть свечка находится строго внутри пирога.

Формат выходных данных

Вывести одно вещественное число с точностью не менее трех знаков после десятичной точки — минимальную разницу между площадями двух получающихся после разрезания кусков, которую сможет получить Малыш.

Примеры

Пример №1

Стандартный ввод
8 5 7 2

Стандартный вывод
12.571

Пример №2

Стандартный ввод
2 2 1 1

Стандартный вывод
0.000

Примечания

На рис. 2.2.5 представлены четыре варианта разделения пирога для первого примера из условия. Можно видеть, что самый близкий к справедливому способ разделения связан с разрезом из левого верхнего угла. Площадь треугольника в этом случае будет равна 96 / 7, площадь четырехугольника равна 184 / 7, и разница равна 88 / 7, что при округлении до трех знаков равно 12,571.

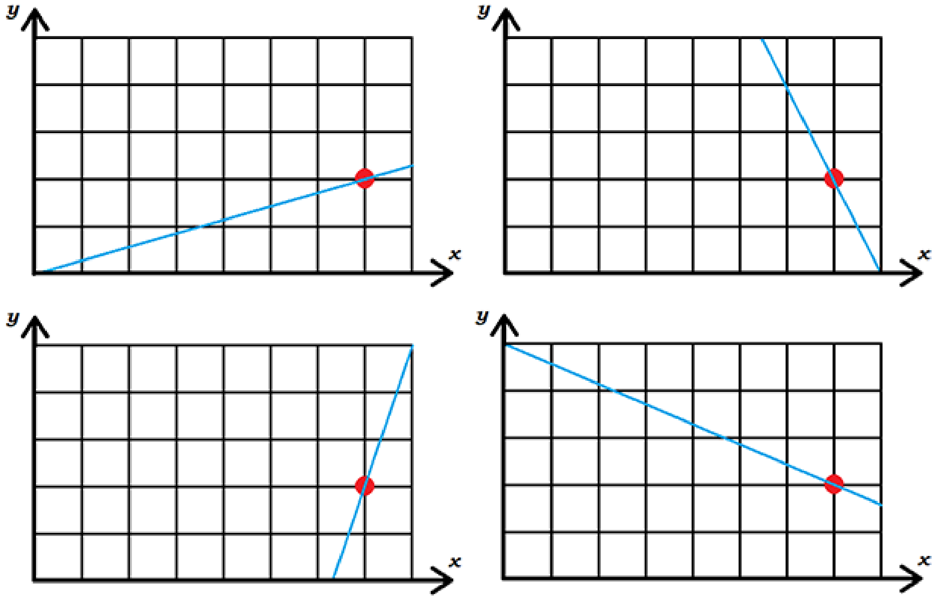

Рис. 2.2.5

Комментарий

Геометрия: для каждого из четырех случаев аккуратно находим катеты прямоугольного треугольника при помощи пропорции, затем находим площадь этого треугольника и, вычитая из всего прямоугольника эту площадь, находим площадь второго куска. Далее выбираем наиболее оптимальное отношение площадей.

Решение

Ниже представлено решение на языке C++.

C++

```cpp
#include<bits/stdc++.h>
#define int long long
using namespace std;
const int INF = 1e18;
double katy(double x, double y, double n){
    return n * y / x;
}
double n, m, x, y;
double ans = INF;
double k1, k2;
void upd(){
    if(k1 < m){
        double st =k1 * n / 2;
        ans = min(ans, n * m - 2 * st);
    }
    else{
        double st =k2 * m / 2;
        ans = min(ans, n * m - 2 * st);
    }}
signed main(){
    cin >> n >> m >> x >> y;
    k1 = katy(x, y, n);
    k2 = katy(y, x, m);
    upd();
    k1 = katy(n - x, y, n);
    k2 = katy(y, n - x, m);
    upd();
    k1 = katy(x, m - y, n);
    k2 = katy(m - y, x, m);
    upd();
    k1 = katy(n - x, m - y, n);
    k2 = katy(m - y, n - x, m);
    upd();
    cout.precision(3);
    cout << fixed << ans<< endl;
}
```

Задача 2.2.3.4. Плетенка (25 баллов)

Имя входного файла: стандартный ввод или `input.txt`.

Имя выходного файла: стандартный вывод или `output.txt`.

Ограничение по времени выполнения программы: 1 с.

Ограничение по памяти: 64 Мбайт.

Условие

У Маши есть n полосок бумаги. i-я полоска имеет ширину 1 и длину a_i. Маша разделит эти полоски на две части и покрасит некоторые в желтый, а оставшиеся — в зеленый цвет. Она сама выберет, какие полоски как покрасить. Далее она хочет из этих полосок сплести максимально большую плетенку. Она расположит полоски одного цвета в некотором порядке горизонтально, а полоски другого цвета в некотором порядке вертикально. После этого она переплетет горизонтальные и вертикальные полоски так, что они будут чередоваться то сверху, то снизу, образуя в местах пересечения шахматную раскраску. Наконец, она обрежет выступающие края полосок так, что останется прямоугольная плетенка с ровными краями. Каждая клетка полученной плетенки должна иметь два слоя.

Маша хочет сплести максимально большую по площади прямоугольную плетенку. Подскажите ей, плетенку какой площади она сможет сделать. Заметим, что она может при создании плетенки использовать не все имеющиеся у нее полоски.

Формат входных данных

В первой строке на вход подается число n — количество полосок бумаги у Маши, где $2 \leqslant n \leqslant 2 \cdot 10^5$. Во второй строке через пробел заданы n целых чисел a_i через пробел — длины полосок, где $1 \leqslant a_i \leqslant 10^9$.

Формат выходных данных

Вывести одно число — площадь прямоугольника, форму которого может иметь самая большая плетенка Маши.

Примеры

Пример №1

Стандартный ввод
8 3 6 5 4 4 5 5 2

Стандартный вывод
12

Примечания

На рис. 2.2.6 представлен один из вариантов получения самой большой плетенки для полосок из примера. Синим обозначена граница полученной максимальной плетенки. Ее размер 3×4, и ее площадь 12. При ее создании Маша не должна использовать полоску номер 8, по этой причине неважно, как она раскрашена.

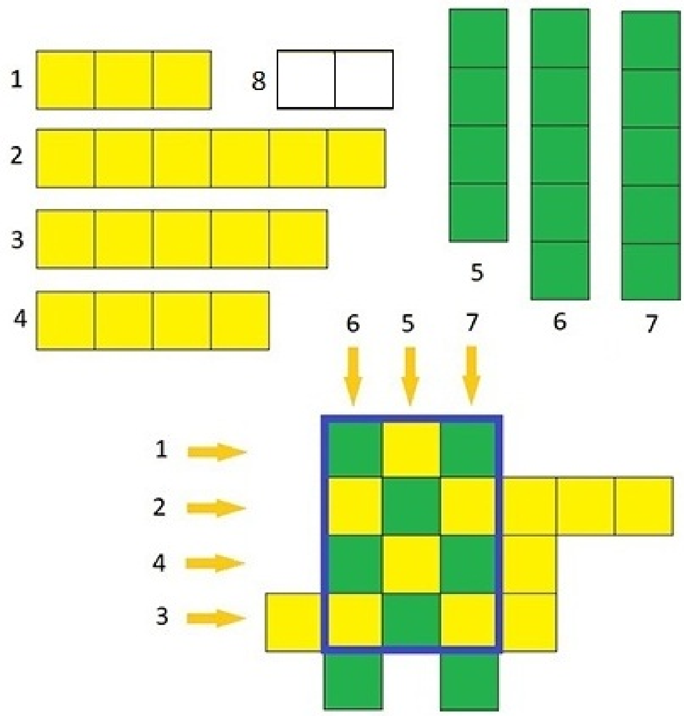

Рис. 2.2.6

Решение

Ниже представлено решение на языке C++.

C++

```cpp
#include<bits/stdc++.h>
#define int long long
using namespace std;
signed main(){
    int n;
    cin >> n;
    deque<int> v(n);
    for(int i = 0; i < n; i++){
        cin >> v[i];
    }
    sort(v.begin(), v.end());
    int ans = 0;
    int cnth = 0, minh;
    while(1){
        if(v.size() == 0){
            break;
        }
        cnth++;
        minh = v.back();
        v.pop_back();
        while(v.size() > 0 && v[0] < cnth){
            v.pop_front();
        }
        ans = max(ans, cnth * min(minh, (int)v.size()));
    }
    cout << ans << endl;
}
```

Задача 2.2.3.5. Английский в игровой форме (30 баллов)

Имя входного файла: стандартный ввод или `input.txt`.

Имя выходного файла: стандартный вывод или `output.txt`.

Ограничение по времени выполнения программы: 3 с.

Ограничение по памяти: 64 Мбайт.

Условие

Маша и Витя запоминают слова английского языка в оригинальной игровой форме. За день им нужно выучить n слов, где $20 \leqslant n \leqslant 100$, каждое из которых имеет длину от 5 до 8 символов. Маша выбирает из этого набора наугад несколько попарно различных слов (также от 5 до 8) и собирает их в одну строку без пробелов. Далее она переставляет буквы в этой строке так, что слова оказываются полностью перепутанными, и дает эту строку Вите. Теперь Витя должен восстановить все слова, которые выбрала Маша.

Но у Вити плохо получается, а Маша уже забыла, какие слова она выбрала. Нужно им помочь — написать программу, которая восстановит слова, выбранные Машей.

Формат входных данных

В первой строке находится строка, которую Маша предложила Вите. Во второй строке содержится число n — количество слов, которые нужно выучить детям, $20 \leqslant n \leqslant 100$.

В следующих n строках содержатся эти слова по одному в строке. Все слова в этом наборе различны. Слова отсортированы в лексикографическом (алфавитном) порядке. Все слова состоят из маленьких букв от a до z. Обратите внимание, что в тестах к этой задаче все заданные слова реально существуют в английском языке и случайным образом выбраны из словаря.

Гарантируется, что длина каждого слова из предложенного набора (словаря) в пределах от 5 до 8, строка, которую получила Маша, может быть получена путем перестановки букв некоторых различных слов из предложенного словаря, причем, набор выбранных Машей слов определяется по ней однозначно. Количество слов, из которых составлена Машина строка, находится в пределах от 5 до 8.

Формат выходных данных

Вывести все слова, выбранные Машей, в алфавитном порядке по одному в строке.

Примеры

Пример №1

Стандартный ввод
stirbaexsudueoeidgomttcrnrwlunapntetacwri 24 bridge cranky document drawing farmer fighter figurine gravy havoc minimum reactant reply republic sonata soprano split subset tailor texture tomorrow trout vicinity wrist writer

Стандартный вывод
document drawing republic sonata texture wrist

Комментарий

В случае, выделенном в условии (слова являются случайными, взятыми из английского словаря), задача решается рекурсией с перебором вариантов.

Решение

Ниже представлено решение на языке C++.

C++

```cpp
#include<bits/stdc++.h>
#define int long long
using namespace std;
string frs;
int n;
vector<string> dict;
vector<int> msk(26, 0);
int cnt = 0;
vector<vector<int> > amsk;
vector<string> ans;
bool bigok = 0;
void p(int pos){
    if(!bigok){
        if(cnt == 0){
            sort(ans.begin(), ans.end());
            bigok = 1;
            return;
        }
        for(int i = pos; i < n; i++){
            string ts = dict[i];
            bool ok = 1;
            for(int j = 0; j < 26; j++){
                if(amsk[i][j] > msk[j]){
                    ok = 0;
                }
            }
            if(ok){
                ans.push_back(ts);
                for(int j = 0; j < 26; j++){
                    msk[j] -= amsk[i][j];
                    cnt -= amsk[i][j];
                }
                p(i + 1);
                if(!bigok){
                for(int j = 0; j < 26; j++){
                    msk[j] += amsk[i][j];
                    cnt += amsk[i][j];
                }
                ans.pop_back();
                }
            }
        }
    }
}
signed main(){
    cin >> frs;
    cin >> n;
    amsk.resize(n, vector<int>(26, 0));

    string ts;
    for(int i = 0; i < n; i++){
        cin >> ts;
        dict.push_back(ts);
    }
    for(int i = 0; i < n; i++){
        for(auto el : dict[i]){
            amsk[i][el - 'a']++;
        }
    }
```

```
    for(auto el : frs){
        msk[el - 'a']++;
        cnt++;
    }
    p(0);
    for(auto el : ans){
        cout << el << endl;
    }
}
```

2.2.4. Четвертая волна. Задачи 8–11 класса

Задачи четвертой волны предметного тура по информатике открыты для решения. Соревнование доступно на платформе Яндекс.Контест: https://contest.yandex.ru/contest/63457/enter/.

Задача 2.2.4.1. Квадратный флаг (10 баллов)

Имя входного файла: стандартный ввод или `input.txt`.

Имя выходного файла: стандартный вывод или `output.txt`.

Ограничение по времени выполнения программы: 1 с.

Ограничение по памяти: 64 Мбайт.

Условие

Одному портному заказали сделать одноцветный флаг. Особенность этого флага в том, что он должен быть квадратным. У портного есть два прямоугольных куска ткани заданного цвета. Один из них имеет размеры $a \times b$, другой — $c \times d$. Так как клиент будет платить пропорционально площади изготовленного флага, портной хочет сначала сшить имеющиеся у него прямоугольные куски, соединив их двумя какими-то сторонами, а затем из полученного полотна вырезать и сделать флаг с максимально большой стороной. Определить сторону получившегося у него флага.

Формат входных данных

На вход подаются две строки. В первой строке находятся размеры первого прямоугольника — целые числа a, b через пробел, во второй — размеры второго прямоугольника, также целые числа c, d через пробел, где $1 \leqslant a, b, c, d \leqslant 10^9$.

Формат выходных данных

Вывести одно число — сторону самого большого квадрата, который можно получить по условию задачи.

Примеры

Пример №1

Стандартный ввод
2 4 3 6

Стандартный вывод
4

Пример №2

Стандартный ввод
2 2 3 6

Стандартный вывод
3

Примечания

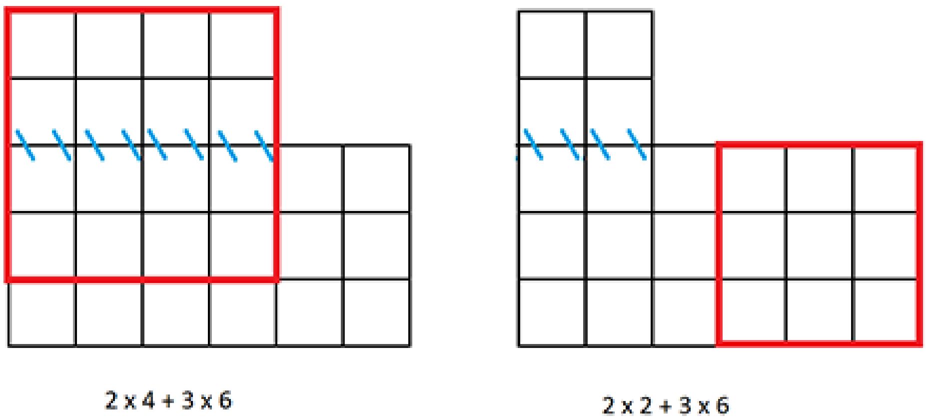

Рис. 2.2.7

На рис. 2.2.7 представлены иллюстрации для тестов из условия. Синими штрихами обозначено место сшивки двух кусков. Красный квадрат выделяет один из вариантов вырезания максимального квадрата.

Решение

Ниже представлено решение на языке C++.

C++

```cpp
#include<bits/stdc++.h>
#define int long long
using namespace std;
signed main(){
    int a, b, c, d;
    cin >> a >> b >> c >> d;
    int ans = max(min(a, b), min(c, d));
    int p1 = min(a + c, min(b, d));
    int p2 = min(a + d, min(b, c));
    int p3 = min(b + c, min(a, d));
    int p4 = min(b + d, min(a, c));
    ans = max({ans, p1, p2, p3, p4});
    cout << ans << endl;
}
```

Задача 2.2.4.2. Потерянная ДНК (15 баллов)

Имя входного файла: стандартный ввод или `input.txt`.

Имя выходного файла: стандартный вывод или `output.txt`.

Ограничение по времени выполнения программы: 1 с.

Ограничение по памяти: 64 Мбайт.

Условие

В данной задаче будем упрощенно считать, что ДНК представляется строкой длины от 10 до 100, состоящей из букв A, C, G, T.

Пусть даны две ДНК D_1 и D_2 одной и той же длины n. Выберем некоторое произвольное число i от 1 до $n-1$ и поменяем местами префиксы (начала) этих ДНК длины i. Будем говорить, что полученные новые две строки образованы путем скрещивания двух исходных по префиксу длины i.

Например, пусть D_1 = **AACGGTAGGT**, а D_2 = TCCCGGAACA. Выберем $i = 4$ и поменяем местами префиксы длины 4. Получим две новые ДНК, одна из которых будет иметь вид **AACG**GGAACA, а вторая — TCCC**GTAGGT**. Для наглядности были выделены части первой из них.

Полученные новые ДНК снова могут быть скрещены по любому префиксу длины от 1 до $n-1$.

Теперь можно рассмотреть популяцию из нескольких ДНК. Выберем из них две, произведем их скрещивание по префиксу какой-либо длины и поместим две новые ДНК в исходную популяцию. В данной задаче будем считать, что количество ДНК не увеличивается, то есть старые две ДНК заменяются на новые две ДНК.

Дана исходная популяция из m ДНК, каждая имеет одну и ту же длину n. После некоторого количества попарных скрещиваний была получена новая популяция. Но при итоговой обработке данных сведения об одной ДНК из новой популяции были потеряны. Задача состоит в отыскании этой потерянной ДНК по оставшимся $m-1$ ДНК из новой популяции.

Формат входных данных

В первой строке через пробел даны два числа n — длина ДНК и m — количество ДНК в исходной популяции, где $10 \leqslant n \leqslant 100$, $2 \leqslant m \leqslant 100$.

В следующих m строках содержится описание исходной популяции ДНК, каждая задается строкой длины n, состоящей из символов A, C, G и T.

Далее следует разделяющая строка, содержащая n символов «—».

Далее следует еще $m - 1$ строк, описывающих новую (заключительную) популяцию без одной ДНК.

Гарантируется, что данные верны, то есть $m - 1$ последняя ДНК является некоторой новой популяцией ровно без одной ДНК, полученной из исходной популяции, заданной в m первых строках.

Формат выходных данных

Вывести недостающую утерянную ДНК.

Примеры

Пример №1

Стандартный ввод

```
10 2
AACGGTAGGT
TCCCGGAACA
----------
TCCCGTAGGT
```

Стандартный вывод

```
AACGGGAACA
```

Пример №2

Стандартный ввод

```
10 4
AACCGGTTAA
ACGTACGTAC
AAACCCGGGT
CATTACTGGA
----------
AAGCGCTTAA
CCACACGTGC
AACTAGGGGT
```

Стандартный вывод

```
AATTCCTGAA
```

Комментарий

Для каждой позиции нужно найти недостающую букву из первого набора ДНК. Для этого удобнее всего использовать функцию `xor`.

Решение

Ниже представлено решение на языке C++.

C++

```cpp
#include<bits/stdc++.h>
#define int long long
using namespace std;
signed main(){
    int n, m;
    cin >> n >> m;
    vector<string> v1(m);
    for(int i = 0; i < m; i++){
        cin >> v1[i];
    }
    string d;
    cin >> d;
    vector<string> v2(m - 1);
    for(int i = 0; i < m - 1; i++){
        cin >> v2[i];
    }
    for(int j = 0; j < n; j++){
        int ss = 0;
        for(int i = 0; i < m; i++){
            ss ^= (int)(v1[i][j]);
        }
        for(int i = 0; i < m - 1; i++){
            ss ^= (int)(v2[i][j]);
        }
        cout << (char)(ss);
    }
    cout << endl;
}
```

Задача 2.2.4.3. Утомленные туристы (20 баллов)

Имя входного файла: стандартный ввод или `input.txt`.

Имя выходного файла: стандартный вывод или `output.txt`.

Ограничение по времени выполнения программы: 1 с.

Ограничение по памяти: 64 Мбайт.

Условие

Рассмотрим следующий вариант известной задачи на перемещение по туннелю группы из четырех человек. В общем виде она выглядит так: четыре туриста хотят пройти по темному туннелю. Имеется один фонарик. По туннелю можно перемещаться либо вдвоем, либо по одному, при этом у тех, кто движется в туннеле, должен быть фонарик в руках. По этой причине движение должно быть следующим: двое переходят туда, один возвращается обратно и приносит фонарик тем, кто еще не перешел. После этого указанный маневр повторяется снова.

У каждого участника своя скорость движения в туннеле. Пусть участники проходят туннель за A, B, C и D мин. Если идут двое, то они движутся со скоростью того, кто идет медленнее. Требуется по заданным временам прохождения туннеля каждого из участников перевести их максимально быстро через туннель.

Немного усложним данную задачу. Введем фактор усталости. А именно, любой участник, пройдя по туннелю, устает и в следующий раз идет уже медленнее. После каждого прохождения туннеля время прохождения любого участника увеличивается на E мин. Например, если участник до начала движения проходит туннель за 1 мин, а показатель усталости E равен 3 мин, то первый раз участник пройдет туннель за 1 мин, второй раз — за 4 мин, третий раз — за 7 мин и т. д.

По заданным A, B, C, D и E узнать, за какое минимальное время можно провести всю группу через туннель согласно указанным правилам.

Формат входных данных

На вход подаются пять чисел. В первой строке через пробел четыре числа A, B, C и D — время прохождения туннеля каждым из четырех участников до того, как они начали движение. Во второй строке содержится число E — величина, на которую увеличивается время прохождения туннеля каждым участником после каждого перемещения. При этом $1 \leqslant A, B, C, D \leqslant 1\,000$, $0 \leqslant E \leqslant 1\,000$.

Формат выходных данных

Вывести одно число — минимальное время прохождения туннеля всей группой.

Примеры

Пример №1

Стандартный ввод
8 9 10 1 3

Стандартный вывод
44

Пример №2

Стандартный ввод
8 9 10 1 0

Стандартный вывод
29

Примечания

В первом примере при прохождении туннеля каждый турист устает и движется медленнее на 3 мин. Покажем, как перевести группу при этом за 44 мин.

Каждую ситуацию будем обозначать следующим образом: слева от двоеточия находятся туристы, которые стоят в начале туннеля, а справа — те, что стоят в конце туннеля. Туриста будем обозначать при помощи числа, соответствующего его текущему времени прохождения туннеля.

Тогда исходная ситуация имеет вид $1, 8, 9, 10 :$.

Сначала идут туристы 1 и 8, каждый после перехода устает на 3 мин, получим ситуацию $9, 10 : 4, 11$, затрачено 8 мин.

Обратно возвращается турист 4, он устает еще на 3 мин. Ситуация становится $7, 9, 10 : 11$, затрачено $8 + 4 = 12$ мин.

Теперь идут туристы 7 и 9, получится ситуация $10 : 10, 11, 12$, затрачено $8 + 4 + 9 = 21$ мин.

Возвращается турист 10, получится $10, 13 : 11, 12$, затрачено $8 + 4 + 9 + 10 = 31$ мин.

Наконец, оставшиеся двое туристов 10 и 13 за 13 мин переходят туннель, итого затрачено $8 + 4 + 9 + 10 + 13 = 44$ мин.

Комментарий

Задача решается рекурсивным перебором всех вариантов прохождения.

Решение

Ниже представлено решение на языке C++.

C++

```cpp
#include<bits/stdc++.h>
#define int long long
using namespace std;
const int INF = 1e18;
vector<int> v(4);
int e, ans = INF;
void p(vector<int> &vl, vector<int> &vr, int tv){
    if(vl.size() == 2){
        ans = min(ans, tv + *max_element(vl.begin(), vl.end()));
        return;
    }
    for(int i = 0; i < vl.size() - 1; i++){
        for(int j = i + 1; j < vl.size(); j++){
            vector<int> vl1;
            for(int k = 0; k < vl.size(); k++){
                if(k != i && k != j){
                    vl1.push_back(vl[k]);
                }
            }
            vector<int> vr1 = vr;
```

```
            vr1.push_back(vl[i] + e);
            vr1.push_back(vl[j] + e);
            int tmp = max(vl[i], vl[j]);
            sort(vr1.rbegin(), vr1.rend());
            vl1.push_back(vr1.back() + e);
            vr1.pop_back();
            p(vl1, vr1, tv + tmp + vl1.back() - e);
        }
    }
}
signed main(){
    for(int i = 0; i < 4; i++){
        cin >> v[i];
    }
    sort(v.begin(), v.end());
    cin >> e;
    vector<int> vl = v, vr;
    p(vl, vr, 0);
    cout << ans;
}
```

Задача 2.2.4.4. Проектируем мост (25 баллов)

Имя входного файла: стандартный ввод или `input.txt`.

Имя выходного файла: стандартный вывод или `output.txt`.

Ограничение по времени выполнения программы: 1 с.

Ограничение по памяти: 64 Мбайт.

Условие

При постройке моста используются два типа пролетов: П-образные (они прочные, но дорогие) и Т-образные (они дешевле, но менее надежные). Мост должен начинаться и заканчиваться П-образными пролетами. Любой Т-образный пролет должен иметь хотя бы один П-образный пролет в качестве соседнего.

Длина проектируемого моста — n пролетов. Муниципалитет выделил средства на постройку a П-образных и b Т-образных пролетов. При этом $a + b = n$. Требуется выяснить, сколькими способами при этих условиях можно скомпоновать мост. Два способа компоновки моста отличаются, если в одной на некоторой позиции стоит П-образный пролет, а в другой на этой же позиции стоит Т-образный пролет.

Формат входных данных

В одной строке через пробел заданы два числа: a — число П-образных пролетов и b — число Т-образных пролетов, на постройку которых выделены средства, где $2 \leqslant a \leqslant 10^6$, $0 \leqslant b \leqslant 10^6$.

Формат выходных данных

Вывести одно число — количество вариантов компоновки моста. Так как ответ может быть очень большим, требуется вывести остаток от его деления на $1\,000\,000\,007$ $(10^9 + 7)$.

Примеры

Пример №1

Стандартный ввод
`4 3`

Стандартный вывод
`7`

Примечания

Для примера из условия имеется 7 вариантов компоновки моста (пробелы добавлены для лучшего восприятия вариантов):

П Т Т П Т П П

П Т Т П П Т П

П Т П Т Т П П

П Т П П Т Т П

П П Т П Т Т П

П П Т Т П Т П

П Т П Т П Т П

Комментарий

При заданных ограничениях задача решается только при помощи комбинаторики с вычислениями по модулю.

Решение

Ниже представлено решение на языке C++.

C++

```
#include<bits/stdc++.h>
#define int long long
using namespace std;
const int INF = 1e18;
const int MOD = 1e9 + 7;
vector<int> f(2e6 + 1, 1);
```

```
int binpow (int a, int n) {
        int res = 1;
        while (n > 0) {
                if (n % 2 == 1)
                        (res *= a) %= MOD;
                (a *= a) %= MOD;
                n /= 2;
        }
        return res;
}

int bc(int n, int k){
    int res = f[n];
    int p1 = binpow(f[k], MOD - 2);
    int p2 = binpow(f[n - k], MOD - 2);
    (res *= p1) %= MOD;
    (res *= p2) %= MOD;
    return res;
}
signed main(){
    for(int i = 1; i <= 2e6; i++){
        f[i] = (f[i - 1] * i) % MOD;
    }
    int a, b;
    int ans = 0;
    cin >> a >> b;
    a--;
    for(int i = 0; i < a + 1; i++){
        if(2 * i <= b){
            int d = bc(a, i);
            if(b - 2 * i <= a - i){
                (d *= bc(a - i, b - 2 * i) ) %= MOD;
                (ans += d) %= MOD;
            }
        }
    }
    cout << ans << endl;
}
```

Задача 2.2.4.5. Джентльмены на прогулке (30 баллов)

Имя входного файла: стандартный ввод или `input.txt`.

Имя выходного файла: стандартный вывод или `output.txt`.

Ограничение по времени выполнения программы: 8 с.

Ограничение по памяти: 64 Мбайт.

Условие

По прямому участку улицы, которую будем считать отрезком AB длины d, прогуливаются n джентльменов. i-й джентльмен движется со скоростью v_i. Скорости всех джентльменов попарно различны. Дойдя до любого конца улицы, каждый джентльмен поворачивает и идет в обратную сторону.

При каждой встрече два джентльмена приветствуют друг друга, приподнимая головной убор. Приветствие происходит и в том случае, когда один джентльмен обгоняет другого. Если два джентльмена встречаются в момент их одновременного поворота, то происходит два приветствия: одно до поворота, другое — после поворота. Если происходит одновременная встреча трех и более джентльменов, то они приветствуют друг друга попарно, то есть каждый каждого. Допустим, если одновременно встретились четыре джентльмена где-то посреди улицы, произойдет шесть попарных приветствий. Если же эти четыре джентльмена встретились в момент их одновременного поворота, произойдет уже двенадцать приветствий.

В этой задаче считаем, что все действия происходят без остановок, то есть и повороты и приветствия происходят мгновенно. Джентльмены одновременно начинают свою прогулку из точки A в момент 0. В этот момент они уже производят свои первые попарные приветствия, то есть в момент 0 уже произведено $n \cdot (n-1)/2$ приветствий. Момент старта не считается моментом поворота, то есть на старте число приветствий не удваивается. Джентльмены гуляют достаточно долго, чтобы произошло любое заданное количество приветствий.

Требуется найти момент, в который было произведено k-е по порядку приветствие.

Формат входных данных

В первой строке ввода через пробел содержится два целых числа: d — длина отрезка AB и n — количество прогуливающихся джентльменов, где $1 \leqslant d \leqslant 200$, $2 \leqslant n \leqslant 2\,000$.

Во второй строке находятся n целых чисел v_i через пробел — скорости каждого джентльмена, где $1 \leqslant v_i \leqslant 2\,000$. Гарантируется, что все скорости попарно различны. Скорости даны в порядке возрастания, то есть $v_1 < v_2 < \ldots < v_n$.

В третьей строке содержится одно целое число k — номер требуемого приветствия, для которого нужно найти момент, когда оно произойдет, где $1 \leqslant k \leqslant 10^9$.

Формат выходных данных

Вывести одно вещественное число — время, когда произойдет k-е по порядку приветствие. Ответ вывести с точностью не менее двух знаков после десятичной точки.

Примеры

Пример №1

Стандартный ввод
5 4 2 5 8 10 6

Стандартный вывод
0.000

Пример №2

Стандартный ввод
`5 4` `2 5 8 10` `7`

Стандартный вывод
`0.556`

Пример №3

Стандартный ввод
`5 4` `2 5 8 10` `11`

Стандартный вывод
`1.000`

Пример №4

Стандартный ввод
`5 4` `2 5 8 10` `15`

Стандартный вывод
`1.429`

Пример №5

Стандартный ввод
`5 4` `2 5 8 10` `17`

Стандартный вывод
`1.667`

Пример №6

Стандартный ввод
`5 4` `2 5 8 10` `19`

Стандартный вывод
`1.667`

Пример №7

Стандартный ввод
5 4 2 5 8 10 21

Стандартный вывод
2.000

Примечания

На рис. 2.2.8 приведено положение джентльменов из примеров в моменты времени 0, 1 и 2. Джентльмены обозначены своими скоростями. Стрелками обозначены направления их движения в соответствующий момент. Перечислим и пронумеруем в порядке возрастания моменты попарных приветствий этих джентльменов до момента времени 2 включительно. Если два и более приветствия происходят одновременно, неважно какое из них конкретно имеет номер k, главное, что они происходят в один и тот же определенный момент времени.

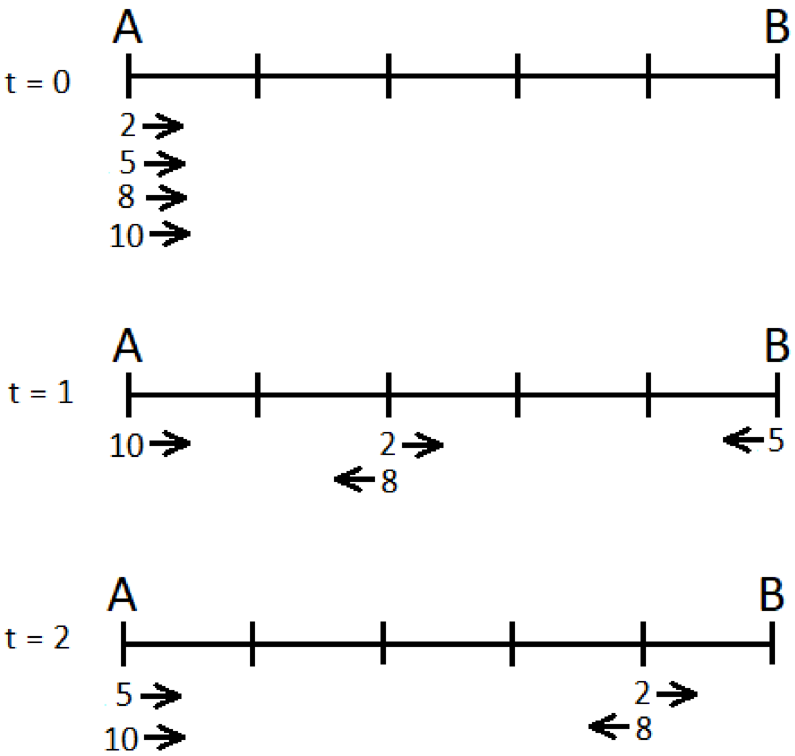

Рис. 2.2.8

1. 2 и 5 приветствуют друг друга в момент 0 (изображено на рис. 2.2.8).
2. 2 и 8 приветствуют друг друга в момент 0 (изображено на рис. 2.2.8).
3. 2 и 10 приветствуют друг друга в момент 0 (изображено на рис. 2.2.8).
4. 5 и 8 приветствуют друг друга в момент 0 (изображено на рис. 2.2.8).
5. 5 и 10 приветствуют друг друга в момент 0 (изображено на рис. 2.2.8).

6. 8 и 10 приветствуют друг друга в момент 0 (изображено на рис. 2.2.8).
7. 8 и 10 приветствуют друг друга в момент 0.556.
8. 5 и 10 приветствуют друг друга в момент 0.667.
9. 5 и 8 приветствуют друг друга в момент 0.769.
10. 2 и 10 приветствуют друг друга в момент 0.833.
11. 2 и 8 приветствуют друг друга в момент 1.000 (изображено на рис. 2.2.8).
12. 8 и 10 приветствуют друг друга в момент 1.111.
13. 2 и 10 приветствуют друг друга в момент 1.250.
14. 5 и 10 приветствуют друг друга в момент 1.333.
15. 2 и 5 приветствуют друг друга в момент 1.429.
16. 5 и 8 приветствуют друг друга в момент 1.538.
17. 2 и 8 приветствуют друг друга в момент 1.667.
18. 2 и 10 приветствуют друг друга в момент 1.667.
19. 8 и 10 приветствуют друг друга в момент 1.667 (в момент 1.667 встретятся одновременно три джентльмена 2, 8 и 10).
20. 2 и 8 приветствуют друг друга в момент 2.000 (изображено на рис. 2.2.8).
21. 5 и 10 приветствуют друг друга в момент 2.000 (до поворота).
22. 5 и 10 приветствуют друг друга в момент 2.000 (после поворота, изображено на рис. 2.2.8).

Комментарий

Задача решается при помощи бинпоиска с квадратичным нахождением ответа в каждой его итерации.

Решение

Ниже представлено решение на языке C++.

C++

```cpp
#include<bits/stdc++.h>
#define int long long
using namespace std;
const double EPS = 1e-7;
double x(double M, int V, int d){
    double dst = V * M;
    int cnt = floor((dst + EPS) / d);
    double pin = dst - cnt * d;
    if(cnt % 2 == 0){
        return pin;
    }
    else{
        return d - pin;
    }
}
int F(double M, vector<int> &v, int d){
    int res = 0;
    for(int i = 0; i < v.size(); i++){
        double dst = v[i] * M;
```

```
            int cnt = floor((dst + EPS) / d);
            res += cnt * i;
            double tx = x(M, v[i], d);
            for(int j = 0; j < i; j++){
                double txj = x(M, v[j], d);
                if(cnt % 2 == 0){
                    res += txj <= tx + EPS;
                }
                else{
                    res += txj >= tx - EPS;
                }
            }
        }
        return res;
    }
signed main(){
    int d, n;
    cin >> d >> n;
    vector<int> v(n);
    for(int i = 0; i < n; i++){
        cin >> v[i];
    }
    int k;
    cin >> k;
    double L = 0, R = 1;
    while(F(R, v, d) <= k){
        R *= 2;
    }
    R *= 2;
    while(R - L > 1e-4){
        double M = (R + L) / 2.0;
        if(F(M, v, d) < k){
            L = M;
        }
        else{
            R = M;
        }
    }
    cout.precision(10);
    cout << fixed << L << endl;
}
```

2.3. Предметный тур. Физика

2.3.1. Первая волна. Задачи 8–9 класса

Задачи первой волны предметного тура по по физике за 8–9 класс открыты для решения. Соревнование доступно на платформе Яндекс.Контест: https://contest.yandex.ru/contest/63463/enter/.

Задача 2.3.1.1. Калориметр (10 баллов)

Условие

Внутренний стакан калориметра представляет собой цилиндр с радиусом $R = 8$ см и высотой $3R$. Внешний стакан также имеет форму цилиндра, стенки которого (как боковые, так и торцы) отстоят от стенок внутреннего на расстояние R. Какая масса теплоизоляционного материала с плотностью $\rho = 25$ кг/м^3 необходима, чтобы полностью заполнить пространство между стаканами?

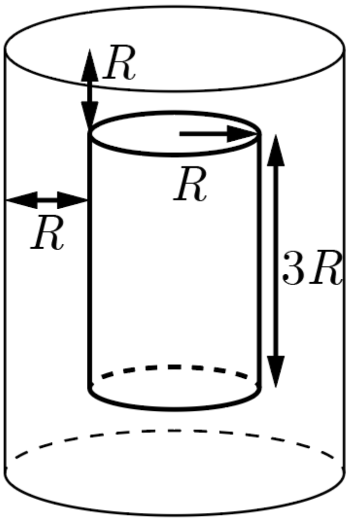

Рис. 2.3.1

Решение

Чтобы найти массу m теплоизоляционного материала, необходимо его плотность умножить на занимаемый им объем V:

$$m = \rho V. \tag{2.3.1}$$

Объем области, заполняемой теплоизоляцией, удобнее всего найти, вычтя из объема V_1 большого внешнего стакана объем V_2 маленького внутреннего. Для любого кругового цилиндра с высотой h и радиусом r объем может быть найден по формуле $V = \pi r^2 h$. В случае большого цилиндра $r = 2R$ и $h = 5R$, следовательно,

$$V_1 = 5R \cdot \pi(2R)^2 = 20\pi R^3. \tag{2.3.2}$$

Аналогично, для маленького $r = R$, $h = 3R$ и, следовательно,

$$V_2 = 3\pi R^3. \tag{2.3.3}$$

Подставляя (2.3.2), (2.3.3) в (2.3.1), получим, что искомая масса составляет:

$$m = \rho(V_1 - V_2) = 17\pi R^3 \rho \approx 0{,}68\,\text{кг}. \tag{2.3.4}$$

Погрешность 0,01 кг.

Ответ: $m = 17\pi R^3 \rho = (0{,}68 \pm 0{,}01)$ кг.

Задача 2.3.1.2. Нить накала (15 баллов)

Условие

Нити накала ламп изготавливают из вольфрама, удельное сопротивление которого сильно зависит от температуры. По мере прогрева нити оно возрастает от $\rho_0 = 5{,}5 \cdot 10^{-8}$ Ом · м до $\rho_1 = 1{,}1 \cdot 10^{-6}$ Ом · м. Определите электрическую мощность, потребляемую лампой в первый момент после ее включения, если в рабочем режиме (полностью прогревшись) лампа потребляет от той же сети мощность $P_1 = 30$ Вт.

Решение

Электрическая мощность P, потребляемая нитью накала, может быть вычислена по закону Джоуля – Ленца, который для фиксированного напряжения U в сети удобно записать как

$$P = U^2/R. \tag{2.3.5}$$

При этом сопротивление R нити легко выразить через ее удельное сопротивление ρ, длину l и площадь поперечного сечения S

$$R = \frac{\rho l}{S}. \tag{2.3.6}$$

Подставляя (2.3.6) в (2.3.5), получим

$$P = \frac{U^2 S}{\rho l},$$

где только ρ зависит от температуры. В результате придем к выводу, что выделяющаяся в лампе мощность обратно пропорциональна ее удельному сопротивлению, откуда окончательно следует

$$P_0 = P_1 \frac{\rho_1}{\rho_0} \approx 600\,\text{Вт}.$$

Погрешность 1 Вт.

Ответ: $P_0 = (600 \pm 1)$ Вт.

Задача 2.3.1.3. Свая (20 баллов)

Условие

Бетонная свая высотой $h = 1{,}4\,\text{м}$ и массой $m = 160\,\text{кг}$ полностью погружена в грунт так, что ее верхний торец совпадает с уровнем почвы. К сожалению, сваю понадобилось извлечь. Определите, какую работу для этого необходимо совершить, если сила трения со стороны грунта, действующая на сваю, прямо пропорциональна площади соприкосновения ее боковой стороны с землей и в начальный момент ее извлечения равна $F = 4\,\text{кН}$. Ускорение свободного падения $g \approx 9{,}8\,\text{м/с}^2$.

Решение

Работа A, необходимая для извлечения сваи, складывается из увеличения потенциальной энергии сваи на величину mgh и работы по преодолению силы трения $A_{\text{тр}}$. Последняя должна быть найдена с учетом постепенного уменьшения силы трения $F_{\text{тр}}$ от максимального значения F до нуля. Поскольку это уменьшение происходит линейно, общая работа оказывается строго вдвое меньше, чем при постоянном значении $F_{\text{тр}} = F$ (аналогично тому, как вычисляется значение работы сил упругости пружины). В результате

$$A = mgh + \frac{Fh}{2} \approx 5\,\text{кДж}. \tag{2.3.7}$$

Погрешность $0{,}1\,\text{кДж}$.

Ответ: $(5{,}0 \pm 0{,}1)\,\text{кДж}$.

Задача 2.3.1.4. Двое из ларца (25 баллов)

Условие

Два дрона одновременно вылетают с общей пусковой станции и движутся по прямолинейным траекториям. Первый дрон на начальном этапе движения перемещается с постоянной скоростью $v_1 = 15\,\text{м/с}$, а через время $\tau = 40\,\text{с}$ быстро переключается на движение с постоянной скоростью $v_2 = 20\,\text{м/с}$. Второй дрон — наоборот, сначала движется со скоростью v_2, а через время τ переключается на скорость v_1. Наконец, дроны одновременно заканчивают полет. Определите, как долго длился этот полет, если по его итогам средняя путевая скорость первого дрона оказалась на $\Delta v = 1\,\text{м/с}$ выше, чем средняя путевая скорость второго.

Решение

По определению средняя путевая скорость — это отношение общего пройденного пути S к общему времени движения t:

$$v = \frac{S}{t}. \tag{2.3.8}$$

Для первого дрона это уравнение принимает вид

$$v_a = \frac{v_1\tau + v_2(t-\tau)}{t}, \tag{2.3.9}$$

а для второго, соответственно,

$$v_b = \frac{v_2\tau + v_1(t-\tau)}{t}. \tag{2.3.10}$$

Из условий задачи известно, что $v_a - v_b = \Delta v$. Подставляя в это уравнение формулы (2.3.9) и (2.3.10), а также домножая его на t, получим

$$v_1\tau + v_2(t-\tau) - v_2\tau - v_1(t-\tau) = \Delta v t. \tag{2.3.11}$$

Перегруппировав слагаемые, получим

$$v_2 t - 2v_2\tau - v_1 t + 2v_1\tau = \Delta v t, \tag{2.3.12}$$

откуда окончательно

$$t = \frac{2(v_2 - v_1)\tau}{v_2 - v_1 - \Delta v} = 100\,\text{с}. \tag{2.3.13}$$

Погрешность 1 с.

Ответ: (100 ± 1) с.

Задача 2.3.1.5. Призма (30 баллов)

Условие

Для тонкого контроля параметров призмы используется следующая установка: отмечается точка, в которую падает лазерный луч, направленный на экран строго под прямым углом (пунктирный на рисунке). Затем на пути луча устанавливается исследуемая призма так, что задняя (первая по ходу распространения луча) ее грань оказывается строго перпендикулярна лучу, и измеряется расстояние d, на которое в результате этого смещается пятно лазера.

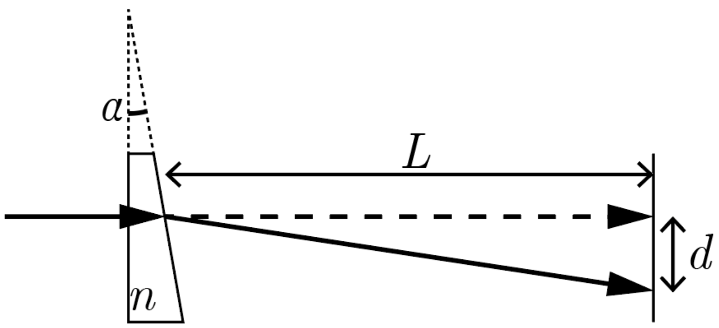

Рис. 2.3.2

Определите показатель преломления стекла, из которого изготовлена призма, если расстояние от передней грани призмы до экрана $L = 3$ м, смещение пятна при установке призмы $d = 12$ см, а угол между передней и задней поверхностями призмы $\alpha = 3°$. Используйте приближение малых углов.

Решение

На первой по ходу распространения луча грани призмы свет не преломляется, поскольку падает на нее под прямым углом. Следовательно, угол падения луча на переднюю грань призмы равен α. Тогда по закону Снеллиуса

$$n = \frac{\sin\beta}{\sin\alpha}, \tag{2.3.14}$$

где β — угол преломления луча, что с учетом приближения малых углов $\sin\alpha \approx \alpha \approx \approx \operatorname{tg}\alpha$ (в радианах) принимает форму

$$n \approx \frac{\beta}{\alpha}. \tag{2.3.15}$$

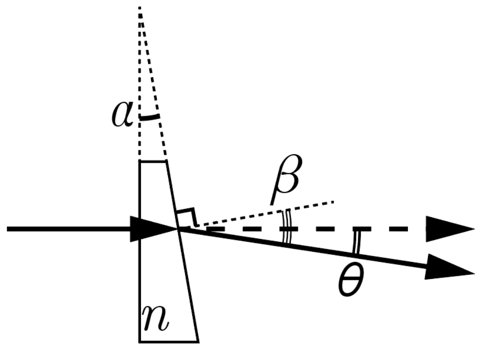

Рис. 2.3.3

Из геометрии рисунка легко видеть, что $\beta = \alpha + \theta$, где θ — угол, который преломленный луч составляет с направлением своего распространения до установки призмы. При этом $\operatorname{tg}\theta = \frac{d}{L}$, откуда

$$n\alpha \approx \beta = \alpha + \operatorname{arctg}\frac{d}{L} \Rightarrow n \approx 1 + \frac{d}{\alpha L} \approx 1{,}76. \tag{2.3.16}$$

Погрешность 0,02.

Ответ: $1{,}76 \pm 0{,}02$.

2.3.2. Первая волна. Задачи 10–11 класса

Задачи первой волны предметного тура по физике за 10–11 класс открыты для решения. Соревнование доступно на платформе Яндекс.Контест: https://contest.yandex.ru/contest/63480/enter/.

Задача 2.3.2.1. Беспилотник (10 баллов)

Условие

Беспилотник, двигаясь равномерно и прямолинейно и обладая при этом импульсом $p_0 = 10^4$ кг · м/с, преодолевает дистанцию $L = 20$ км ровно за 1,5 мин. За какое время преодолеет ее этот же беспилотник, двигаясь также равномерно и прямолинейно, но обладая на $\Delta p = 10^3$ кг · м/с меньшим импульсом?

Решение

Импульс p тела — это произведение его массы на его скорость, поэтому скорость беспилотника легко может быть вычислена как

$$p_0 = \frac{mL}{t_0}, \tag{2.3.17}$$

где t_0 — время в пути с импульсом p_0. Искомое время t можно аналогично выразить через скорость v беспилотника во второй рассмотренной ситуации как

$$t = \frac{L}{v} = \frac{Lm}{p_0 - \Delta p}. \tag{2.3.18}$$

Выражая массу беспилотника из 2.3.17 и подставляя ее в 2.3.18, получим:

$$t = \frac{\cancel{L} p_0 t_0}{\cancel{L}(p_0 - \Delta p)} = 100\,\text{с}. \tag{2.3.19}$$

Погрешность 1 с.

Ответ: (100 ± 1) с.

Задача 2.3.2.2. Грузовик (15 баллов)

Условие

На плоское горизонтальное дно кузова транспортного грузовика погрузили большой грузовой контейнер и забыли его закрепить. Благодаря силе трения контейнер оставался в покое относительно грузовика до тех пор, пока ускорение последнего не превосходило $a_0 = 2{,}0\,\text{м/с}^2$, и начинал скользить при превышении этого значения. Совершая маневр, грузовик приобрел ускорение $a = 2{,}12\,\text{м/с}^2$, направленное по ходу движения. Какое время длился маневр, если в результате контейнер сдвинулся на $l = 1{,}5\,\text{м}$ относительно грузовика?

Решение

Исходя из того, что контейнер остается на месте при ускорении грузовика до a_0, можно, по второму закону Ньютона, заключить, что сила трения покоя, обеспечивающая это ускорение для контейнера, не превышает значения

$$F = ma_0, \tag{2.3.20}$$

где m — масса контейнера. При любом маневре, при котором контейнер начинает скользить, на него действует сила трения скольжения, равная F и, следовательно, его ускорение относительно дороги оказывается равно a_0.

Во время маневра ускорение контейнера относительно грузовика равно (по модулю) $a - a_0$, а пройденное контейнером относительно грузовика расстояние может быть выражено по законам кинематики как

$$l = \frac{(a - a_0)t^2}{2}, \tag{2.3.21}$$

где t — искомое в задаче время. Преобразуя эту формулу, получим

$$t = \sqrt{\frac{2l}{a - a_0}} = 5\,\text{с}. \tag{2.3.22}$$

Погрешность 0,1 с.

Ответ: $(5{,}0 \pm 0{,}1)$ с.

Задача 2.3.2.3. Методичка (20 баллов)

Условие

На лабораторной работе по физике в распоряжении школьников оказались резисторы двух номиналов: с сопротивлениями x и y кОм, а также конденсаторы двух номиналов: с емкостями x и y мкФ, при этом $x > y$. В старой методичке, посвященной этой лабораторной работе, была изображена схема, приведенная на рисунке, чернила на которой сильно затерлись. В результате Витя решил, что изображенные элементы являются резисторами, и, соединив их согласно схеме, получил элемент с эквивалентным сопротивлением $R = 5\,\text{кОм}$. Таня же решила, что это конденсаторы и, соединив их, получила элемент с эквивалентной емкостью $C = 2\,\text{мкФ}$. Определите, чему равнялось число y.

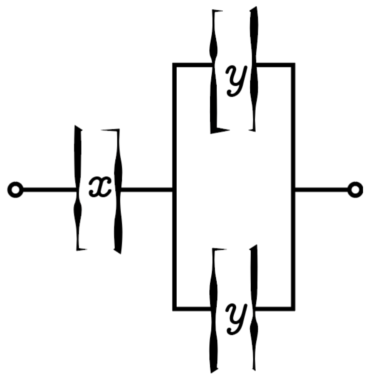

Рис. 2.3.4

Решение

При последовательном соединении резисторов их сопротивления складываются, а при параллельном — складываются обратные сопротивлениям величины. Поэтому сопротивление R схемы Вити через числа x и y в килоомах (кОм) может быть выражено по формуле

$$R = x + \frac{y}{2}. \tag{2.3.23}$$

Для конденсаторов правила поиска эквивалентной емкости при их последовательном и параллельном соединениях в точности обратные, поэтому емкость схемы Тани может быть выражена в микрофарадах (мкФ) по формуле

$$C = \frac{2xy}{x+2y}. \tag{2.3.24}$$

Выразим x из уравнения (2.3.23) и подставим в (2.3.24):

$$C = \frac{2(R-y/2)y}{R+3y/2}. \tag{2.3.25}$$

Домножив полученное уравнение на знаменатель дроби и раскрыв скобки, получим

$$RC + \frac{3Cy}{2} = 2Ry - y^2. \tag{2.3.26}$$

Поскольку величины x и y имеют разную размерность в разных частях задачи, имеет смысл сразу перейти к численным значениям

$$10 + 3y = 10y - y^2. \tag{2.3.27}$$

Это квадратное уравнение, которое легко привести к канонической форме

$$y^2 - 7y + 10 = 0 \tag{2.3.28}$$

и решить с помощью дискриминанта

$$y_{1,2} = \frac{7 \pm \sqrt{49-40}}{2} = \frac{7 \pm 3}{2}. \tag{2.3.29}$$

Снова воспользовавшись уравнением (2.3.23), легко определить, что при $y = 5$ (корень квадратного уравнения с плюсом) $x = 2{,}5$, что не удовлетворяет условию $x > y$. В то же время, при $y = 2$ (корень с минусом) $x = 4$, что удовлетворяет этому условию. Стало быть, верный корень — $y = 2$.

Погрешность 0,1.

Ответ: $2{,}0 \pm 0{,}1$.

Задача 2.3.2.4. Морозилка (25 баллов)

Условие

Морозильная установка работает по циклу Карно, обходимому в обратном направлении. Какую работу должна потребить такая установка, чтобы заморозить $m = 0{,}4$ кг воды, взятой при ее температуре замерзания, передав полученную от нее теплоту в помещение, температура θ которого равна 30 °C? Удельная теплота плавления и кристаллизации воды $\lambda = 333$ кДж/кг, абсолютный ноль температур $T_0 = -273$ °C.

Решение

Идеальный тепловой двигатель (машина Карно) работает, как известно, по циклу, состоящему из двух изотерм и двух адиабат, и имеет КПД

$$\eta = \frac{T_2 - T_1}{T_2}, \tag{2.3.30}$$

где T_1 — минимальная, а T_2 — максимальная температуры в цикле. По определению КПД тепловой машины он равен отношению работы A, совершаемой газом за цикл, к теплоте Q_2, получаемой им от нагревателя

$$\frac{T_2 - T_1}{T_2} = \eta = \frac{A}{Q_2}. \tag{2.3.31}$$

Отсюда Q_2 может быть выражено как

$$Q_2 = \frac{AT_2}{T_2 - T_1}. \tag{2.3.32}$$

Поскольку за полный цикл внутренняя энергия не изменяется, количество теплоты Q_1, которую газ отдает холодильнику такой машины, равна разнице

$$Q_1 = Q_2 - A = \frac{AT_1}{T_2 - T_1}. \tag{2.3.33}$$

В рассматриваемой задаче тепловой двигатель заменен холодильной машиной, для чего его цикл необходимо обходить в обратном направлении. При этом тепловой резервуар с температурой T_2 начинает получать тепло, а с температурой T_1 — отдавать, но по модулю количества теплоты, которыми газ обменивается с этими тепловыми резервуарами, не изменяются. Остается заметить, что в описанной в условиях холодильной машине $T_1 = 0\,^\circ\text{C} = 273\,\text{K}$, $T_2 = \theta$, $Q_1 = \lambda m$, поскольку забираемая у теплового резервуара с меньшей температурой теплота идет на замораживание в нем воды. Подставив эти данные в (2.3.33), получим

$$\lambda m = \frac{AT_1}{\theta - T_1}, \tag{2.3.34}$$

откуда окончательно выразим ответ

$$A = \frac{\lambda m(\theta - T_1)}{T_1} \approx 14{,}6\,\text{Дж}. \tag{2.3.35}$$

Погрешность: 0,5 Дж.

Ответ: $(14{,}6 \pm 0{,}5)$ Дж.

Задача 2.3.2.5. Электромагнит (30 баллов)

Условие

Для большого промышленного электромагнита критически важной стала проблема охлаждения. Было установлено, что при пропускании через электромагнит тока $I_1 = 10$ А он нагревается до температуры $t_1 = 70\,°\mathrm{C}$, после чего перестает увеличивать свою температуру, а при пропускании через него тока $I_2 = 20$ А — до температуры $t_2 = 205\,°\mathrm{C}$.

Определите температуру θ в помещении цеха, в котором используется электромагнит, если известно, что основным механизмом, отвечающим за охлаждение магнита, выступает теплопроводность, мощность которой прямо пропорциональна разнице температур между телами, обменивающимися теплом.

Решение

Как указано в условиях, мощность теплопроводности прямо пропорциональна разнице температур между магнитом и окружающим его воздухом в помещении цеха. Обозначим коэффциент этой пропорциональности κ

$$\begin{cases} P_1 = \kappa(\theta - t_1), \\ P_2 = \kappa(\theta - t_2). \end{cases} \tag{2.3.36}$$

Повышение температуры останавливается, когда мощность производимого катушкой тепла и мощность тепла, уходящего от катушки, благодаря теплообмену, оказываются равны. Первую можно выразить из закона Джоуля – Ленца

$$\begin{cases} P_1 = I_1^2 R, \\ P_2 = I_2^2 R, \end{cases} \tag{2.3.37}$$

где R — сопротивление катушки.

Разделим друг на друга уравнения системы (2.3.36) и уравнения системы (2.3.37), а затем приравняем эти отношения:

$$\frac{P_1}{P_2} = \frac{\theta - t_1}{\theta - t_2} = \frac{I_1^2}{I_2^2}. \tag{2.3.38}$$

Тривиальными алгебраическими преобразованиями выразим θ

$$(\theta - t_1)I_2^2 = (\theta - t_2)I_1^2 \Rightarrow \theta = \frac{t_1 I_2^2 - t_2 I_1^2}{I_2^2 - I_1^2} = 25\,°\mathrm{C}. \tag{2.3.39}$$

Погрешность: $0{,}1\,°\mathrm{C}$.

Ответ: $(25{,}0 \pm 0{,}1)$.

2.3.3. Вторая волна. Задачи 8–9 класса

Задачи второй волны предметного тура по физике за 8–9 класс открыты для решения. Соревнование доступно на платформе Яндекс.Контест: https://contest.yandex.ru/contest/63464/enter/.

Задача 2.3.3.1. Аккумулятор тепла (10 баллов)

Условие

Для печи отопления требуется разработать аккумулятор тепла, представляющий собой емкость фиксированного объема, заполненную тем или иным минералом. Используя таблицу 2.3.1 плотностей ρ и удельных теплоемкостей $c_{\text{уд}}$ различных подходящих для этого пород, расположите их в порядке увеличения количества теплоты, которое может быть запасено в таком аккумуляторе при его нагреве до одной и той же температуры θ. Считайте, что θ заведомо меньше температур, при которых любой из этих минералов начнет плавиться или химически разрушаться, а тепловое расширение этих минералов при нагреве до θ пренебрежимо мало.

Таблица 2.3.1. Плотности и удельные теплоемкости

	Минерал	ρ, г/см^3	$c_{\text{уд}}$, кДж/(кг $\cdot$ °C)
A	Кварц	2,6	0,75
B	Базальт	2,8	0,85
C	Талькохлорит	2,75	0,98
D	Нефрит	3	1,1
E	Порфирит	1,45	0,83

Введите в поле ответа последовательность букв, соответствующих выбранным минералам, без пробелов, от наименьшего к наибольшему количеству запасаемой теплоты.

Решение

Количество тепла Q, которое может быть запасено в тепловом аккумуляторе фиксированного объема V, удобно выразить через массу m материала этого аккумулятора

$$Q = c_{\text{уд}} m(\theta - t_0), \tag{2.3.40}$$

где t_0 — начальная температура теплоаккумулятора. В свою очередь, масса m элементарно выражается через плотность вещества и объем V

$$m = \rho V, \tag{2.3.41}$$

откуда

$$Q = c_{\text{уд}} \rho V(\theta - t_0). \tag{2.3.42}$$

Поскольку величины V, θ, t_0 независимы от выбранного вещества, задача сводится к расположению в порядке возрастания произведений $c_{\text{уд}}\rho$. Найдем эти произведения для всей таблицы 2.3.1.

Таблица 2.3.2

	Минерал	ρ, г/см^3	$c_{уд}$, кДж/(кг · °C)	$c_{уд}\rho$, кДж/(м^3 · °C)
A	Кварц	2,6	0,75	1 950
B	Базальт	2,8	0,85	2 380
C	Талькохлорит	2,75	0,98	2 695
D	Нефрит	3	1,1	3 300
E	Порфирит	1,45	0,83	1 204

Ответ: EABCD.

Задача 2.3.3.2. Изображения (15 баллов)

Условие

Тонкая собирающая линза имеет фокусное расстояние $F = 20$ см. Вдоль ее оптической оси перед линзой расположено плоское зеркало, на расстоянии $h = 2$ см от которого и $d = 60$ см от линзы размещен светодиод S (рис. 2.3.5). Найдите расстояние между двумя действительными изображениями светодиода, формируемыми этой оптической системой.

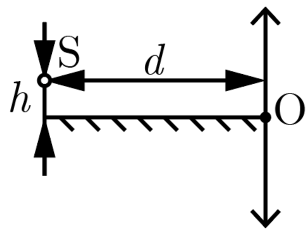

Рис. 2.3.5

Решение

Зеркало формирует мнимое изображение S_1 источника, расположенное в противоположном от него полупространстве на таком же расстоянии от зеркала, как и сам источник. В силу перпендикулярности зеркала и линзы, это мнимое изображение также окажется на расстоянии d от плоскости линзы. Далее линза формирует два действительных изображения: одно непосредственно от источника S (на рис. 2.3.6 оно обозначено S_2) и другое — от его мнимого изображения S_1 (S_3 на рисунке).

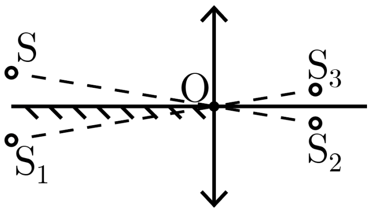

Рис. 2.3.6

Расстояние f, на котором находятся оба действительных изображения от плоскости линзы, легко найти по формуле тонкой линзы

$$\frac{1}{f} + \frac{1}{d} = \frac{1}{F} \Rightarrow f = \frac{Fd}{d - F}. \tag{2.3.43}$$

Искомое расстояние l между изображениями S_2 и S_3, благодаря подобию треугольников $\triangle \mathrm{OSS_1}$ и $\triangle \mathrm{OS_2S_3}$, относится к расстоянию $2h$ между источником и его мнимым изображением S_1 так же, как расстояния от соответствующих изображений и источников до плоскости линзы, являющиеся высотами указанных треугольников

$$\frac{l}{2h} = \frac{f}{d} = \frac{F}{d - F}. \tag{2.3.44}$$

Отсюда окончательно находим

$$l = 2h\frac{F}{d - F} = 2\,\text{см}. \tag{2.3.45}$$

Погрешность 0,1 см.

Ответ: $l = (2{,}0 \pm 0{,}1)$ см.

Задача 2.3.3.3. Пила (30 баллов)

Условие

Циркулярная пила представляет собой пильный диск диаметром $D = 19$ см, вращающийся с частотой 4 500 об/мин. Определите среднюю силу сопротивления заготовки вращению полотна пилы, если за один пропил, длившийся $t = 3$ с, выделилось $Q = 5$ кДж тепла, а пила соприкасалась с заготовкой только узкой полоской своей внешней кромки.

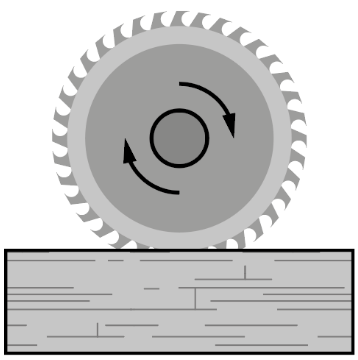

Рис. 2.3.7

Решение

При вращении пилы диссипативные силы (трения различных типов) переводят механическую энергию пильного диска в тепловую. При этом количество выделяющегося тепла равно по модулю работе A этих сил. Последнюю легко найти из ее определения

$$A = FS = Fvt, \tag{2.3.46}$$

где S — путь точек соприкосновения диска с заготовкой, v — скорость этих точек. При вращении диска скорость точек его кромки удобно выразить через период вращения T этого диска

$$v = \frac{\pi D}{T} = \pi D\nu, \tag{2.3.47}$$

где ν — частота вращения диска в оборотах в секунду. Подставляя (2.3.47) в (2.3.46) и учитывая $Q = A$, получим окончательно

$$Q = \pi F D\nu t \Rightarrow F = \frac{Q}{\pi D\nu t} \approx 37\,\text{Н}. \tag{2.3.48}$$

Погрешность 1 Н.

Ответ: $F = (37 \pm 1)\,\text{Н}$.

Задача 2.3.3.4. Питстоп (25 баллов)

Условие

Транспортный робот перемещается из города A в город B, двигаясь практически все время с некоторой постоянной скоростью v. Однако один раз за маршрут ему необходима остановка для заправки и краткого технического обслуживания. Инженеры установили, что при уменьшении длительности этой остановки вдвое скорость движения робота на остальной части маршрута можно будет снизить на $\delta = 10\%$, сохранив при этом его среднюю путевую скорость, что поможет повысить безопасность и экономичность движения. Определите, на сколько процентов (от исходного значения) удалось бы снизить скорость v без изменения средней путевой, если бы от технической остановки удалось полностью отказаться?

Решение

Средняя путевая скорость определяется как отношение всего пути ко всему времени, которое этот путь занимает. Поскольку расстояние между городами неизменно, сохранение средней путевой скорости означает и сохранение общего времени в пути (включая остановку). Следовательно, уменьшение длительности остановки на Δt эквивалентно увеличению времени непосредственного движения на ту же величину. Обозначим общее время робота в пути t, исходную длительность его остановки τ, а исходную скорость движения v_0. Тогда путь робота может быть выражен до и после снижения времени остановки как

$$S = v_0(t - \tau) = v_0(1 - \delta)\left(t - \frac{\tau}{2}\right). \tag{2.3.49}$$

Сократив v_0 и перегруппировав слагаемые, получим

$$\delta t = \tau\left(1 - \frac{1}{2} + \frac{\delta}{2}\right) = \tau\frac{\delta+1}{2}. \qquad (2.3.50)$$

Полностью избавившись от остановки, таким образом, робот будет двигаться в течение времени

$$t = \tau\frac{\delta+1}{2\delta}. \qquad (2.3.51)$$

Аналогично, время движения $t-\tau$ при наличии остановки удобно записать как

$$t - \tau = \tau\left(\frac{\delta+1}{2\delta} - 1\right) = \tau\frac{1-\delta}{2\delta}. \qquad (2.3.52)$$

Обозначив v_1 скорость, которой можно добиться, исключив остановку, запишем через эти выражения путь и приравняем его в случаях с остановкой и без

$$v_1 t = v_0(t-\tau) \Rightarrow v_1\tau\frac{\delta+1}{2\delta} = v_0\tau\frac{1-\delta}{2\delta}. \qquad (2.3.53)$$

Отсюда окончательно

$$\frac{v_1}{v_0} = \frac{1-\delta}{1+\delta} \approx 0{,}818. \qquad (2.3.54)$$

Итого скорость движения без остановки может составлять приблизительно 81,8% от исходной скорости, то есть ниже ее на 18,2%.

Погрешность 0,5%.

Ответ: $(18{,}2 \pm 0{,}5)$%.

Задача 2.3.3.5. Сцепка (30 баллов)

Условие

Два абсолютно одинаковых ползунковых реостата, сопротивления которых могут изменяться в пределах от 0 до $R_0 = 2$ кОм, размещены параллельно на печатной плате и соединены как изображено на рис. 2.3.8. Из-за ошибки в процессе пайки изоляция их ползунков слиплась таким образом, что ползунки всегда занимают одно и то же положение на обоих реостатах, но электрический контакт между ними отсутствует (это соединение обозначено на рисунке пунктиром). Найдите разницу между максимальным и минимальным сопротивлениями полученной батареи.

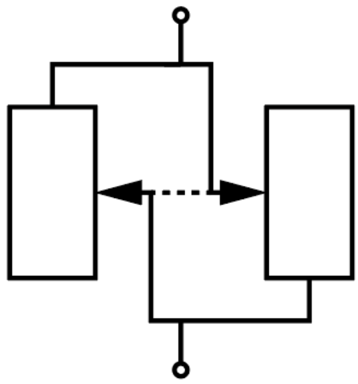

Рис. 2.3.8

Решение

Реостаты на схеме соединены параллельно, поэтому общее сопротивление схемы R может быть выражено через сопротивления $R_{1,2}$ каждого из реостатов по формуле

$$R = \frac{R_1 R_2}{R_1 + R_2}. \tag{2.3.55}$$

Несложно видеть из схемы, что когда ползунок находится в крайнем верхнем положении, левый реостат имеет нулевое сопротивление, а правый — сопротивление R_0 и наоборот. Величина сопротивления находится в линейной зависимости от длины провода. Из этого можно заключить, что при любом положении ползунка

$$R_2 = R_0 - R_1. \tag{2.3.56}$$

Подставляя этот результат в (2.3.55), получим

$$R = \frac{R_1(R_0 - R_1)}{R_0} = R_1 - \frac{R_1^2}{R_0}. \tag{2.3.57}$$

График полученной функции является параболой. Его минимумы и максимумы могут лежать либо на границах диапазона изменения R_1, либо в вершине соответствующей параболы. Поскольку коэффициент перед квадратным слагаемым отрицательный, парабола «повернута» ветвями вниз, то есть на границах диапазона (при $R_1 = 0$ или $R_1 = R_0$) сопротивления батареи минимальны и равны 0, а в ее вершине (которая, как легко видеть из симметрии или непосредственно по формуле $x_{max} = -b/(2a)$, лежит в центре диапазона, при $R_1 = R_2 = \dfrac{R_0}{2}$) сопротивление батареи равно $\dfrac{R_0}{4}$.

Таким образом,

$$R_{max} - R_{min} = \frac{R_0}{4} - 0 = \frac{R_0}{4} = 0{,}5\,\text{кОм}. \tag{2.3.58}$$

Погрешность $0{,}01$ кОм.

Ответ: $l = (0{,}50 \pm 0{,}01)$ кОм.

2.3.4. Вторая волна. Задачи 10–11 класса

Задачи второй волны предметного тура по по физике за 10–11 класс открыты для решения. Соревнование доступно на платформе Яндекс.Контест: https://contest.yandex.ru/contest/63481/enter/.

Задача 2.3.4.1. Зарядка (10 баллов)

Условие

Для увеличения ресурса аккумулятора его зарядка происходит по специальной программе, учитывающей внешние условия, интенсивность использования прибора и другие факторы. Рассчитав оптимальный режим, зарядное устройство в течении $\tau = 10$ мин подавало на аккумулятор ток, линейно возраставший со временем от нуля до максимального значения $I_0 = 3$ А, затем в течение $2{,}5\tau$ поддерживало постоянное значение этого тока и, наконец, на протяжении $\tau/2$, также линейно опускало ток от максимального значения до нуля. Какой общий заряд поступил на положительную клемму аккумулятора за все это время?

Решение

Один из способов решения задачи состоит в обнаружении аналогии между током и движением. Подобно тому, как скорость описывает темп изменения координаты, сила тока описывает темп поступления заряда на аккумулятор. Из кинематики известно, что при равномерном увеличении этого темпа (равноускоренном движении) от нуля, либо при равномерном снижении этого темпа (равнозамедленном движении) до нуля тело проходит вдвое меньшее расстояние, чем при движении с постоянной скоростью, равной максимальной на рассматриваемом участке. Применяя этот результат к току, заметим, что за время $2{,}5\tau$ постоянного тока зарядки на аккумулятор поступил заряд

$$q_1 = 2{,}5\tau I_0, \tag{2.3.59}$$

а за общее время $1{,}5\tau$ увеличения и уменьшения силы тока — заряд

$$q_2 = \frac{1{,}5\tau I_0}{2} = 0{,}75\tau I_0. \tag{2.3.60}$$

Складывая эти заряды, получим окончательно

$$q = 3{,}25\tau I_0 = 5850\,\text{Кл}. \tag{2.3.61}$$

Задача также может быть решена графически, изображением зависимости $I(t)$ и вычислением площади под ней. Фактически такое решение также является применением аналогии, но геометрической, а не кинематической.

Погрешность 50 Кл.

Ответ: (5850 ± 50) Кл.

Задача 2.3.4.2. Принтер (15 баллов)

Условие

Печатающая головка 3D-принтера может перемещаться вдоль направляющей (координата x), которая также может смещаться в перпендикулярном направлении (координата y) под действием двух сервоприводов. Для изготовления детали на принтер была передана программа, согласно которой сервоприводы должны перемещать головку по законам $x(t) = 0{,}2\sin(t/10)$; $y(t) = 0{,}1 + 0{,}2\cos(t/10)$, где t — время, а все величины даны в основных единицах СИ. Найдите величину ускорения печатающей головки при выполнении этой программы.

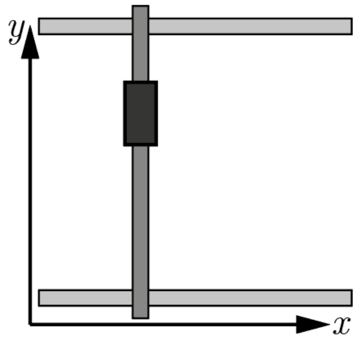

Рис. 2.3.9

Решение

Прежде всего заметим, что уравнения приведенного типа параметрически задают окружность. Это следует из самого определения синуса и косинуса. Радиус R этой окружности равен множителю перед синусом и косинусом (т. к. в математическом определении тригонометрических функций используется единичная окружность), то есть $R = 0{,}2$ м. Здесь было учтено, что основными единицами СИ для измерения длины являются метры.

Поскольку зависимость угла на окружности (аргумента синуса и косинуса) от времени линейна, модуль скорости v печатающей головки постоянен. Чтобы найти его, определим период обращения. Печатающая головка описывает полную окружность, когда аргумент синуса и косинуса меняется на 2π, что происходит при достижении t значения $T = 20\pi$ с. Тогда

$$v = \frac{2\pi R}{T}. \tag{2.3.62}$$

Окончательно заметим, что при неизменной по модулю скорости головки единственное ее ускорение — центростремительное, которое может быть найдено как

$$a = \frac{v^2}{R} = \frac{4\pi^2 R}{T^2} = \frac{4\cancel{\pi^2} \cdot 0{,}2}{400\cancel{\pi^2}} \approx 2\,\text{мм/с}^2. \tag{2.3.63}$$

Погрешность $0{,}1\,\text{мм/с}^2$.

Ответ: $(2{,}0 \pm 0{,}1)\,\text{мм/с}^2$.

Задача 2.3.4.3. Плита (20 баллов)

Условие

Квадратная плита ABCD со стороной $a = 40\,\text{см}$ шарнирно закреплена в одной точке и вращается вокруг оси, перпендикулярной ее плоскости с постоянной угловой скоростью. При этом в некоторый момент времени скорость вершины C этой плиты направлена строго на вершину D, а ускорение вершины A — строго на вершину B (см. рис. 2.3.10). На каком расстоянии от центра пластины находится шарнир?

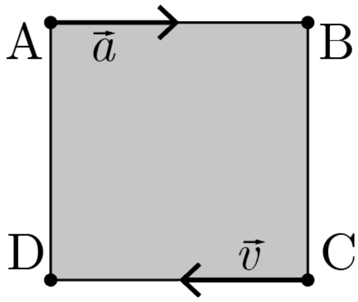

Рис. 2.3.10

Решение

При плоском вращении скорость каждой точки тела направлена перпендикулярно направлению из этой точки на ось вращения, а ускорение (центростремительное) — непосредственно на эту ось. Поэтому, проведя одну прямую через точку C перпендикулярно ее скорости, а другую — через точку A вдоль ее ускорения, можно найти ось вращения как точку пересечения этих прямых. Такой точкой будет, разумеется, вершина B, а расстояние от нее до центра пластины равно

$$l = \frac{\sqrt{2}}{2}a \approx 28{,}3\,\text{см}. \qquad (2.3.64)$$

Погрешность $0{,}5\,\text{см}$.

Ответ: $(28{,}3 \pm 0{,}5)\,\text{см}$.

Задача 2.3.4.4. Прямоугольники (25 баллов)

Условие

К проекту модельного теплового двигателя, рабочим телом которого является идеальный одноатомный газ, прилагается pV-диаграмма его рабочего цикла, представляющая собой прямоугольник 1234. К сожалению, автор не указал ни давления, ни объемы характерных точек, а ограничился «площадями» (в энергетических единицах) некоторых прямоугольников на данной диаграмме, которые указаны на рис. 2.3.10. Да к тому же самая важная площадь — площадь внутри цикла 1234, стерлась. Тем не менее определите КПД данного двигателя.

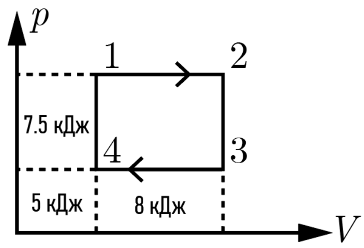

Рис. 2.3.11

Решение

КПД η теплового двигателя определяется как отношение работы A газа за один цикл этого двигателя к количеству теплоты Q, получаемой газом за этот цикл от нагревателя:

$$\eta = \frac{A}{Q}. \tag{2.3.65}$$

Первая — есть площадь прямоугольника 1234. Найти ее просто, если заметить, что поскольку высота (вдоль оси p) у верхних двух прямоугольников одинакова, их площади относятся так же, как и их ширины (вдоль оси V), и то же верно для нижней пары прямоугольников

$$A = 7{,}5\frac{8}{5} = 12\,\text{кДж}. \tag{2.3.66}$$

Чтобы найти теплоту Q, воспользуемся первым началом термодинамики $Q = A + \Delta U$ и заметим, что газ получает теплоту на процессах 41 и 12. Работа за эти два процесса равна полной площади под отрезком 12

$$A_{412} = 12 + 8 = 20\,\text{кДж}, \tag{2.3.67}$$

а внутренняя энергия может быть удобно вычислена по формуле

$$U = \frac{3}{2}PV, \tag{2.3.68}$$

из которой следует, что внутренняя энергия U_4 газа в состоянии 4 равна

$$U_4 = \frac{3}{2}5 = 7{,}5\,\text{кДж}, \tag{2.3.69}$$

поскольку произведение PV для этого состояния есть площадь одного соответствующего прямоугольника. Аналогично, внутренняя энергия U_2 газа в состоянии 2 равна

$$U_2 = \frac{3}{2}(7{,}5 + 12 + 5 + 8) = 48{,}75\,\text{кДж}, \tag{2.3.70}$$

поскольку в этом состоянии соответствующее произведение PV равно общей площади всех прямоугольников на диаграмме.

Подставляя все найденные величины в исходное уравнение (2.3.65), получим окончательно

$$\eta = \frac{A}{A_{412} + U_2 - U_4} = \frac{12}{20 + 48{,}75 - 7{,}5} \approx 19{,}6\%. \tag{2.3.71}$$

Погрешность 1%.

Ответ: $(19{,}6 \pm 1{,}0)\%$

Задача 2.3.4.5. Трюм (30 баллов)

Условие

Трюм грузового судна представляет собой призму с основанием в виде равностороннего треугольника вершиной вниз. Длина внешней стороны треугольника $a = 15$ м, толщина его стенок $d = 1{,}5$ м, длина киля судна (высота призмы) $l = 40$ м. Инженерами было рассчитано, что для сохранения устойчивости судна центр тяжести сыпучего груза, перевозимого в трюме, должен быть хотя бы на $h = 2$ м ниже центра тяжести вытесняемой трюмом воды при его полном погружении. Какой максимальный объем груза можно разместить в трюме, если при насыпании его центр тяжести занимает самое низкое доступное положение?

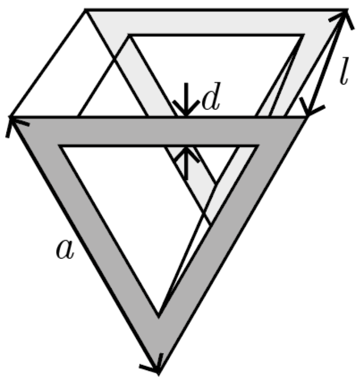

Рис. 2.3.12

Решение

Сыпучий груз занимает объем, форма которого также является призмой с основанием в виде равностороннего треугольника, во всяком случае, при необходимости понизить центр тяжести. В силу симметрии, центр тяжести равностороннего треугольника находится в его геометрическом центре. Поскольку центр тяжести груза должен быть на h ниже, чем центр тяжести вытесненной воды, центр треугольника, формируемого сечением насыпанного груза (темный на рис. 2.3.13), на h ниже центра трюма.

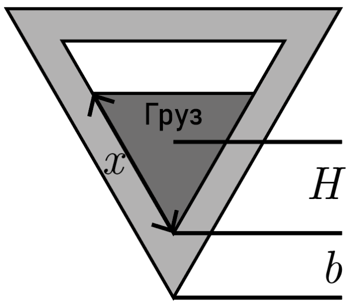

Рис. 2.3.13

Отсчитывая от киля (нижней вершины треугольника) высоту $y_в$, на которой находится центр тяжести вытесненной воды, легко выразить как

$$y_{\text{в}} = \frac{\sqrt{3}}{3}a, \tag{2.3.72}$$

поскольку радиус описанной окружности для равностороннего треугольника в $\sqrt{3}/3$ раз меньше его стороны.

Высота $y_{\text{г}}$ центра тяжести груза может быть найдена как

$$y_{\text{г}} = b + H, \tag{2.3.73}$$

где b — толщина стенки вдоль соответствующего направления (см. рис. 2.3.13), а H — высота центра тяжести груза от нижней внутренней точки трюма. Обе эти величины вычисляются из тригонометрии

$$b = 2d; \quad H = \frac{\sqrt{3}}{3}x, \tag{2.3.74}$$

где x — сторона треугольника, являющегося поперечным сечением груза.

Учитывая $y_{\text{г}} + h = y_{\text{в}}$, получим

$$\frac{\sqrt{3}}{3}x + 2d + h = \frac{\sqrt{3}}{3}a, \tag{2.3.75}$$

откуда

$$x = a - \sqrt{3}(2d + h) \approx 6{,}34\,\text{м}. \tag{2.3.76}$$

Объем, занимаемый грузом при такой длине его стороны, находится как произведение площади равностороннего треугольника на длину киля

$$V = \frac{\sqrt{3}}{4}x^2 l \approx 696\,\text{м}^3. \tag{2.3.77}$$

Погрешность 10 м^3.

Ответ: (696 ± 10) м^3.

2.3.5. Третья волна. Задачи 8–9 класса

Задачи третьей волны предметного тура по физике за 8–9 класс открыты для решения. Соревнование доступно на платформе Яндекс.Контест: https://contest.yandex.ru/contest/63465/enter/.

Задача 2.3.5.1. Башня (10 баллов)

Условие

В гидравлической системе используется башня, заполненная минеральным маслом с плотностью $\rho = 900\,\text{кг/м}^3$. Верхний уровень масла расположен на $h = 60\,\text{м}$ выше, чем смотровое окно в трубе с маслом, закрепленное на ней при помощи $n = 16$ одинаковых болтов. Определите, какую нагрузку должен выдерживать каждый из этих болтов на разрыв, чтобы обеспечить трехкратный запас прочности? Площадь смотрового окна $S = 0{,}1\,\text{м}^2$. Ускорение свободного падения $g = 9{,}8\,\text{м/с}^2$.

Решение

Давление p масла на уровне окна элементарно вычисляется по формуле гидростатического давления

$$p = \rho g h. \tag{2.3.78}$$

Исходя из определения всякого давления p как отношения силы F к площади S, на которую действует эта сила, найдем силу со стороны жидкости, «пытающуюся выдавить» смотровое окно

$$F = pS = \rho g h S. \tag{2.3.79}$$

Искомая рассчетная нагрузка f каждого из болтов может быть получена домножением этой силы на 3 (требуемый запас прочности) и делением на число болтов n, по которым распределяется нагрузка

$$f = \frac{3F}{16} = \frac{3\rho g h S}{16} \approx 9{,}9\,\text{кН}. \tag{2.3.80}$$

Погрешность $0{,}1\,$кН.

Ответ: $(9{,}9 \pm 0{,}1)\,$кН.

Задача 2.3.5.2. Реостат (15 баллов)

Условие

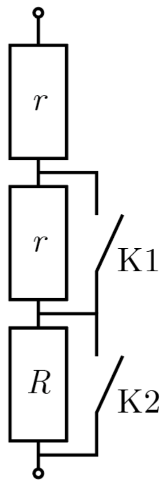

Рис. 2.3.14

Имея в своем распоряжении резисторы только двух различных номиналов, начинающий радиолюбитель изготовил ступенчатый реостат оригинальной конструкции, позволяющий, переключая ключи, получить четыре различных значения сопротивления. Увы, на приложенной к прибору схеме он забыл указать сопротивления отдельных резисторов, указав только, какие из них совпадают. В техническом паспорте устройства остались данные о том, что при замыкании ключа K_1 и размыкании ключа K_2 оно имеет сопротивление $R_1 = 2$ кОм, а напротив, при замыкании ключа K_2 и размыкании ключа K_1 — сопротивление $R_2 = 3$ кОм. Найдите максимальное сопротивление, которое можно получить, используя этот реостат.

Решение

Когда параллельно резистору коротко замыкается цепь, этот резистор фактически перестает работать, поскольку сопротивление провода пренебрежимо мало. Следовательно, замыкание ключа К1 фактически эквивалентно замене среднего резистора на отрезок провода, а ключа К2 — такой же замене нижнего. Учитывая это и применяя формулу эквивалентного сопротивления последовательно соединенных резисторов, легко получим

$$\begin{cases} R_1 = r + R, \\ R_2 = 2r. \end{cases} \tag{2.3.81}$$

Решая эту систему, находим $r = \frac{R_2}{2}$ и $R = R_1 - \frac{R_2}{2}$. Теперь точно так же составим выражения для оставшихся двух конфигураций реостата: R_3 с обоими замкнутыми ключами и R_4 с обоими разомкнутыми

$$\begin{cases} R_3 = r = \dfrac{R_2}{2} = 1{,}5\,\text{кОм}, \\ R_4 = 2r + R = R_1 + \dfrac{R_2}{2} = 3{,}5\,\text{кОм}. \end{cases} \tag{2.3.82}$$

Погрешность $0{,}01\,\text{кОм}$.

Ответ: $(3{,}50 \pm 0{,}1)\,\text{кОм}$.

Задача 2.3.5.3. Теплоноситель (20 баллов)

Условие

В некоторых типах ядерных реакторов в качестве теплоносителя используются жидкие металлы. Определите, сколько теплоты за 1 с забирает у реактора жидкий свинец с удельной теплоемкостью $c = 155$ Дж/(кг·°С) и средней плотностью $\rho = 10^4$ кг/м^3, если, двигаясь в трубе диаметром $d = 10$ см со скоростью $v = 20$ м/с, он нагревается от $t_1 = 400\,°\text{C}$ до $t_2 = 900\,°\text{C}$.

Решение

Обозначим рассматриваемый промежуток времени (1 с) τ. Двигаясь со скоростью v, свинец успевает за это время пройти в трубе дистанцию $l = v\tau$. Учитывая площадь сечения трубы $S = \pi d^2/4$, можно заключить из этого, что за время τ в реактор поступает и из реактора уходит объем

$$V = Sl = \frac{\pi}{4} d^2 v\tau \tag{2.3.83}$$

расплавленного свинца. Количество теплоты Q, которое этот свинец забирает у реактора, задается выражением

$$Q = cm(t_2 - t_1) = c\rho V(t_2 - t_1) = \frac{\pi}{4} c\rho d^2 v\tau(t_2 - t_1) \approx 122\,\text{МДж}, \tag{2.3.84}$$

где m — масса поступившей и ушедшей порции свинца.

Погрешность 2 МДж.

Ответ: (122 ± 2) МДж.

Задача 2.3.5.4. Шкала (25 баллов)

Условие

Шкала вольтметра, используемого в эксперименте, имеет вид, представленный на рис. 2.3.15, и общую длину $l = 15$ см (от минимальной до максимальной отметки). Экспериментатор, глядя на прибор под углом 45° к плоскости шкалы, считал показания вольтметра как $U_1 = 1{,}2$ В ровно, однако на самом деле стрелка прибора находилась напротив отметки $U_2 = 0{,}8$ В. Определите, на какое расстояние отстоит стрелка от шкалы, если известно, что глаза экспериментатора находились со шкалой строго на одном уровне высоты, а деления расположены на шкале равномерно.

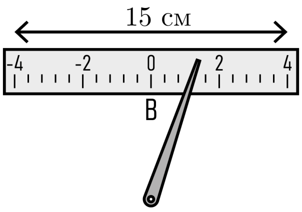

Рис. 2.3.15

Решение

Согласно условиям, глаза экспериментатора находятся на одной высоте со шкалой, поэтому удобно изобразить систему в горизонтальной плоскости (вид сверху), см. рис. 2.3.16. Поскольку угол α, под которым наблюдатель смотрит на стрелку, равен 45°, искомое расстояние x в точности равно расстоянию y между точкой A действительных показаний прибора и точкой B считанных экспериментатором показаний (треугольник ABC, где C — стрелка — прямоугольный и равнобедренный).

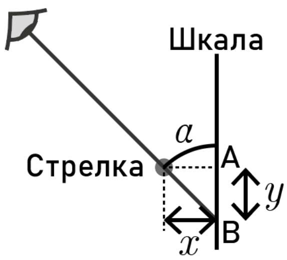

Рис. 2.3.16

Тогда остается вычислить расстояние y между отметками шкалы, соответствующими значениям U_1 и U_2. Поскольку шкала равномерная, это расстояние относится к полной длине шкалы так же, как величина абсолютной ошибки к ее разнице между ее верхним U_{max} и нижним U_{min} пределами измерений

$$\frac{y}{l} = \frac{U_2 - U_1}{U_{max} - U_{min}}. \tag{2.3.85}$$

Отсюда окончательно

$$x = y = l\frac{U_2 - U_1}{U_{max} - U_{min}} = 7{,}5 \text{ мм}. \qquad (2.3.86)$$

Погрешность 0,1 мм.

Ответ: $(7{,}5 \pm 0{,}1)$ мм.

Задача 2.3.5.5. Площадка (30 баллов)

Условие

Три робота одновременно стартуют в углу A прямоугольной площадки ABCD. Все они движутся с постоянными по модулю скоростями и все заканчивают движение в точке D одновременно. Но первый робот движется по прямой вдоль стороны AD, второй — по трехзвенной ломаной ABCD, а третий — по двузвенной: сначала вдоль диагонали AC, а затем — по стороне CD. Во сколько раз средняя путевая скорость третьего робота выше, чем первого, если средняя путевая скорость второго робота выше, чем первого, в 2,5 раза? Временем на разгон, остановку и развороты роботов можно пренебречь.

Решение

Обозначив длину стороны AB (и, соответственно, CD) прямоугольника a, а длину стороны BC (и, соответственно, DA) — b, можно легко выразить через эти стороны пути $S_{1,2,3}$ всех трех роботов

$$\begin{cases} S_1 = b, \\ S_2 = 2a + b, \\ S_3 = \sqrt{a^2 + b^2} + a. \end{cases} \qquad (2.3.87)$$

Поскольку время движения всех роботов совпадало, отношения их путей точно такие же, как и средних путевых скоростей:

$$\frac{v_2}{v_1} = \frac{S_2}{S_1} = \frac{2a}{b} + 1 = 2{,}5, \qquad (2.3.88)$$

откуда легко найти

$$\frac{2a}{b} = 1{,}5 \Rightarrow b = \frac{4}{3}a. \qquad (2.3.89)$$

Теперь, пользуясь той же логикой, найдем ответ на вопрос задачи как отношение путей третьего и первого роботов

$$\frac{v_3}{v_1} = \frac{S_3}{S_1} = \frac{\sqrt{a^2 + b^2} + a}{b} = \frac{\sqrt{\frac{25a^2}{9}} + a}{\frac{4a}{3}} = \frac{\frac{8a}{3}}{\frac{4a}{3}} = 2. \qquad (2.3.90)$$

Погрешность 0,01.

Ответ: $2{,}00 \pm 0{,}01$.

2.3.6. Третья волна. Задачи 10–11 класса

Задачи третьей волны предметного тура по по физике за 10–11 класс открыты для решения. Соревнование доступно на платформе Яндекс.Контест: https://contest.yandex.ru/contest/63482/enter/.

Задача 2.3.6.1. На коленке (10 баллов)

Условие

На конференции, посвященной освоению труднодоступных северных регионов, был представлен проект теплового двигателя, для работы которого используются два тепловых резервуара. Их можно собрать «на коленке»: в качестве холодильника выступает емкость с мокрым снегом, а в качестве нагревателя — котелок с кипящей водой. При этом, по заверениям авторов проекта, КПД двигателя только в $\alpha = 1{,}6$ раз уступает КПД идеальной тепловой машины с такими же холодильником и нагревателем. Найдите этот КПД. Абсолютный ноль температур $T_0 = -273\,^\circ\mathrm{C}$. Двигатель используется при нормальном атмосферном давлении на уровне моря.

Решение

Мокрый снег представляет собой смесь льда и воды, поэтому может существовать только при температуре плавления льда, $t_х = 0\,^\circ\mathrm{C}$. Аналогично, при атмосферном давлении температура кипящей воды может быть равна только $t_н = 100\,^\circ\mathrm{C}$. КПД идеальной тепловой машины (машины Карно) с известными абсолютными термодинамическими температурами $T_н$ и $T_х$ холодильника задается выражением

$$\eta_0 = \frac{T_н - T_х}{T_н}. \tag{2.3.91}$$

Чтобы дать ответ на вопрос задачи, таким образом, остается разделить этот КПД на α и перевести температуры в шкалу Кельвина

$$\eta = \frac{\eta_0}{\alpha} = \frac{t_н - t_х}{\alpha(t_н - T_0)} \approx 16{,}8\%. \tag{2.3.92}$$

Погрешность: 0,2%.

Ответ: $(16{,}8 \pm 0{,}2)\%$.

Задача 2.3.6.2. Патруль (15 баллов)

Условие

Корабль береговой охраны движется с постоянной скоростью $v = 12$ м/с относительно поверхности воды. Наблюдательный дрон запрограммирован летать на постоянной высоте по траектории, в системе отсчета корабля представляющей собой окружность с радиусом $R = 2$ км и центром на этом корабле, двигаясь в этой системе отсчета равномерно и совершая полный оборот за время $T = 10$ мин. Во сколько раз максимальная скорость дрона относительно поверхности воды выше его минимальной скорости относительно нее же?

Решение

Согласно правилу сложения скоростей, скорость $\vec{v}_{\text{дв}}$ дрона относительно воды равна (векторной) сумме его скорости $\vec{v}_{\text{дк}}$ относительно корабля и скорости $\vec{v}_{\text{кв}}$ корабля относительно воды. Поскольку направление вектора $\vec{v}_{\text{кв}}$ неизменно, а вектор $\vec{v}_{\text{дк}}$ в ходе движения дрона принимает все возможные в горизонтальной плоскости направления, обязательно найдутся такие моменты времени, когда эти два вектора сонаправлены и такие, когда они противонаправлены. Эти два случая и будут соответствовать максимальному и минимальному значениям модуля суммы этих векторов, равным $v_{max} = |\vec{v}_{\text{дк}}| + |\vec{v}_{\text{кв}}|$ и $v_{min} = ||\vec{v}_{\text{дк}}| - |\vec{v}_{\text{кв}}||$ соответственно.

Модуль вектора $\vec{v}_{\text{кв}}$ дан напрямую: он равен v. Модуль вектора $\vec{v}_{\text{дк}}$ легко получить, разделив путь дрона в СО корабля на период его обращения в этой СО

$$v_{\text{дк}} = \frac{2\pi R}{T}. \tag{2.3.93}$$

Отсюда получим окончательно

$$\frac{v_{max}}{v_{min}} = \frac{vT + 2\pi R}{|vT - 2\pi R|} \approx 3{,}68. \tag{2.3.94}$$

Погрешность 0,02.

Ответ: $3{,}68 \pm 0{,}02$.

Задача 2.3.6.3. Конденсатор (20 баллов)

Условие

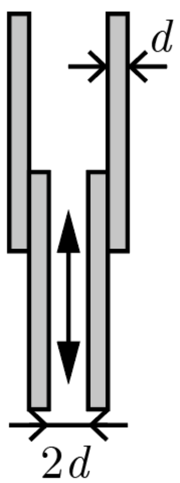

Рис. 2.3.17

Переменный конденсатор состоит из двух пар металлических пластинок толщиной d и площадью $S \gg d^2$, разделенных воздушными зазорами. При этом одна из пар (внутренняя) может частично или полностью входить в зазор другой (внешней), плотно прилегая к ней так, что электрический контакт между соответствующими пластинами никогда не нарушается, но при полном выдвижении площадь этого контакта пренебрежимо мала в сравнении с S.

Определите, во сколько раз максимальная емкость такого конденсатора превосходит минимальную, если зазор между пластинами внутренней пары имеет ширину $2d$.

Решение

Рассматриваемый конденсатор может быть представлен как батарея из двух параллельно соединенных конденсаторов, один из которых (внутренняя пара пластин) всегда имеет зазор $2d$ и площадь обкладок S, а другой (внешняя пара) имеет зазор $4d$ и площадь обкладок, которая может изменяться в пределах от 0 до S. Поскольку эти конденсаторы соединены параллельно, эквивалентная емкость батареи равна сумме их емкостей, а значит, максимальна, когда пары пластин максимально раздвинуты и минимальна, когда они полностью вдвинуты одна в другую. Используя формулу емкости плоского конденсатора, найдем, что емкость внутренней пары пластин (она же минимальная емкость всей батареи) равна

$$C_1 = \frac{S\varepsilon_0}{2d}, \tag{2.3.95}$$

где ε_0 — диэлектрическая постоянная.

Аналогично, емкость полностью выдвинутой внешней пары равна

$$C_2 = \frac{S\varepsilon_0}{4d}, \tag{2.3.96}$$

а максимальная емкость батареи составляет, соответственно, $C_1 + C_2$.

Тогда для искомого отношения максимальной и минимальной емкостей получим:

$$\frac{C_{max}}{C_{min}} = \frac{C_1 + C_2}{C_1} = \frac{\cancel{S\varepsilon_0}/(4\cancel{d}) + \cancel{S\varepsilon_0}/(2\cancel{d})}{\cancel{S\varepsilon_0}/(2\cancel{d})} = \frac{1/4 + 1/2}{1/2} = 1{,}5. \tag{2.3.97}$$

Погрешность 0,01.

Ответ: $1{,}50 \pm 0{,}01$.

Задача 2.3.6.4. Аэростат (25 баллов)

Условие

Горелка теплового аэростата способна поддерживать среднюю температуру воздуха в его оболочке не более, чем на $\Delta t = 70\,^\circ\mathrm{C}$ выше, чем температура окружающего шар воздуха. Аэростат имеет объем $V = 645$ м^3, а общая масса его оболочки, корзины и полезной нагрузки $M = 150$ кг. Определите, при какой максимальной температуре окружающей среды аэростат сможет взлететь?

Абсолютный ноль температур $T_0 = -273\,°\mathrm{C}$, атмосферное давление $p_0 = 100\,\text{кПа}$, молярная масса воздуха $\mu = 29\,\text{г/моль}$, универсальная газовая постоянная $R = 8{,}31\,\text{Дж/(моль} \cdot \text{К)}$.

Решение

Оболочки монгольфьеров (тепловых аэростатов) представляют собой открытые сосуды, поэтому давление внутри и снаружи оболочки должно совпадать (и равняться p_0). Тогда из уравнения Менделеева – Клапейрона

$$p_0 V = \frac{m}{\mu} RT, \tag{2.3.98}$$

где T — абсолютная термодинамическая температура газа, m — его масса.

Легко выразить массу m_1 воздуха внутри оболочки и массу m_2 вытесненного атмосферного воздуха

$$\begin{aligned} m_1 &= \frac{\mu p_0 V}{R(T_0 + \Delta t)}, \\ m_2 &= \frac{\mu p_0 V}{RT_0}, \end{aligned} \tag{2.3.99}$$

где T_0 — искомая температура окружающего воздуха.

Для того чтобы аэростат мог подняться в воздух, необходимо, чтобы вес вытесняемого им воздуха превысил его общий вес (включая вес газа в оболочке)

$$m_2 \cancel{g} = M\cancel{g} + m_1 \cancel{g} \Rightarrow \frac{\mu p_0 V}{RT_0} = M + \frac{\mu p_0 V}{R(T_0 + \Delta t)}, \tag{2.3.100}$$

где g — ускорение свободного падения (сразу сокращающееся во всех слагаемых).

Домножим это выражение на $RT_0(T_0 + \Delta T)$ и получим квадратное уравнение относительно T_0

$$\mu p_0 V(\cancel{T_0} + \Delta t) = MRT_0(T_0 + \Delta t) + \cancel{\mu p_0 V T_0}. \tag{2.3.101}$$

В канонической форме:

$$T_0^2 + T_0 \Delta t - \frac{\mu p_0 V \Delta t}{MR} = 0. \tag{2.3.102}$$

Остается найти (методом дискриминанта) единственный положительный корень этого уравнения

$$T_0 = -\frac{\Delta t}{2} + \frac{1}{2}\sqrt{\Delta t^2 + \frac{4\mu p_0 V}{MR}\Delta t} \approx 291\,\mathrm{K} \approx 18\,°\mathrm{C}. \tag{2.3.103}$$

Погрешность $0{,}1\,°\mathrm{C}$.

Ответ: $(18{,}0 \pm 0{,}1)\,°\mathrm{C}$.

Задача 2.3.6.5. Спутник (30 баллов)

Условие

Спутник, движущийся вокруг Земли по высокой круговой орбите, перевели на другую круговую орбиту, в результате чего его кинетическая энергия увеличилась на 5%. На сколько процентов увеличился модуль потенциальной энергии взаимодействия спутника с планетой, если она считается равной нулю на бесконечном удалении от планеты?

Решение

Обозначим радиус орбиты спутника R, его орбитальную скорость v, его массу m, а массу планеты M. На спутник действует сила всемирного тяготения

$$F = G\frac{mM}{R^2}, \tag{2.3.104}$$

где G — гравитационная постоянная, связанная с центростремительным ускорением v^2/R спутника вторым законом Ньютона

$$\cancel{m}\frac{v^2}{\cancel{R}} = G\frac{\cancel{m}M}{R^{\cancel{2}}}. \tag{2.3.105}$$

Из этого выражения легко видеть, что квадрат орбитальной скорости спутника v^2 обратно пропорционален радиусу орбиты. Разумеется, кинетическая энергия спутника $mv^2/2$ прямо пропорциональна этому квадрату скорости и, следовательно, тоже обратно пропорциональна R.

Чтобы понять, как потенциальная энергия спутника зависит от радиуса его орбиты, проще всего обратить внимание на аналогию между гравитацией и электростатическими силами. Сила Кулона взаимодействия двух точечных зарядов зависит от расстояния между ними и обратно пропорциональна квадрату разделяющего их расстояния, точно так же, как сила всемирного тяготения. Одновременно потенциальная энергия взаимодействия этих зарядов обратно пропорциональна первой степени расстояния между ними, следовательно, то же справедливо и для гравитации. В результате видно, что модуль потенциальной энергии обратно пропорционален R, как и кинетическая энергия. Следовательно, при изменении орбиты спутника он изменится ровно во столько же раз.

Погрешность 0,01%.

Ответ: $(5{,}00 \pm 0{,}01)\%$.

2.3.7. Четвертая волна. Задачи 8–9 класса

Задачи четвертой волны предметного тура по физике за 8–9 класс открыты для решения. Соревнование доступно на платформе Яндекс.Контест: https://contest.yandex.ru/contest/63466/enter/.

Задача 2.3.7.1. Расплав (10 баллов)

Условие

Расплавленная соль предлагается как эффективный аккумулятор тепла для некоторых типов теплоцентралей. Удельная теплота плавления и кристаллизации соли $\lambda = 28{,}7$ кДж/кг, ее теплоемкость в твердой форме $c_1 = 0{,}92$ кДж/(кг°C), а в жидкой — $c_2 = 1{,}5$ кДж/(кг°C), температура ее плавления $\theta = 800\,°\mathrm{C}$. Определите, какую массу соли необходимо взять, чтобы при ее нагреве от $t_0 = 20\,°\mathrm{C}$ до $t_1 = 1200\,°\mathrm{C}$ запасти $Q = 1$ МДж тепла.

Решение

Количество теплоты, требуемое на нагрев твердой соли до температуры плавления, задается выражением

$$Q_1 = c_1 m(\theta - t_0), \tag{2.3.106}$$

где m — масса нагреваемой соли.

Количество теплоты, уходящее непосредственно на плавление

$$Q_2 = \lambda m. \tag{2.3.107}$$

Наконец, количество теплоты, уходящее на нагрев расплава соли,

$$Q_3 = c_2 m(t_1 - \theta). \tag{2.3.108}$$

Складывая все эти порции тепла, получим, что общая запасаемая теплота

$$Q = Q_1 + Q_2 + Q_3 = m(c_1(\theta - t_0) + \lambda + c_2(t_1 - \theta)), \tag{2.3.109}$$

откуда окончательно

$$m = \frac{Q}{(c_1(\theta - t_0) + \lambda + c_2(t_1 - \theta))} \approx 743\,\text{г}. \tag{2.3.110}$$

Погрешность 1 г.

Ответ: (743 ± 1) г.

Задача 2.3.7.2. Катафот (15 баллов)

Условие

Катафот представляет собой два одинаковых квадратных зеркала, соединенных общей гранью под прямым углом друг к другу. При падении на него видимого света каждое зеркало поглощает $\eta = 20\%$ достигающей его световой энергии, а остальную — отражает. Параллельно биссектрисе образованного зеркалами угла в плоскости, перпендикулярной к их общему ребру, на середину одного из зеркал падает узкий лазерный луч, переносящий мощность $P = 5$ мВт. Какое количество световой энергии поглотит второе зеркало за $\tau = 10$ с?

Решение

Прежде всего отметим, что любой луч, падающий на катафот параллельно его биссектрисе, отразится последовательно от обоих зеркал катафота, как изображено на рис 2.3.18. При этом после первого отражения мощность луча снизится в $(1 - \eta)$ раз, и доля η от этой оставшейся мощности будет поглощена вторым зеркалом. В результате связь между изначальной P и поглощаемой P_1 мощностями имеет следующий вид:

$$P_1 = (1 - \eta)\eta P. \tag{2.3.111}$$

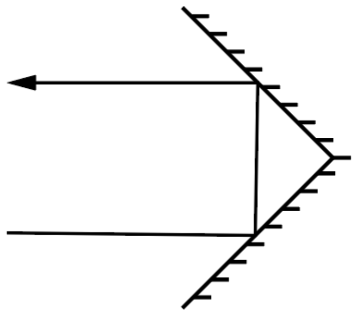

Рис. 2.3.18

Теперь остается лишь вспомнить определение мощности, как отношения энергии (в данном случае переносимой лазерным лучом или поглощаемой зеркалом) ко времени, чтобы получить окончательный ответ

$$Q = P_1\tau = (1 - \eta)\eta P\tau \approx 8\,\text{мДж}. \tag{2.3.112}$$

Погрешность $0{,}1$ мДж.

Ответ: $(8{,}0 \pm 0{,}1)$ мДж.

Задача 2.3.7.3. Соты (20 баллов)

Условие

Композитный материал изготавливают, вырезая из алюминия (плотность $\rho_1 = 2{,}7\,\text{г/см}^3$) строго периодическую вдоль двух взаимно перпендикулярных осей квадратную сетку с толщиной стенки d и длиной внутренней стороны ячейки $4d$, фрагмент которой изображен на рис. 2.3.19. Затем полости заполняют смолой, после затвердевания имеющей плотность $\rho_2 = 1{,}2\,\text{г/см}^3$. Найдите среднюю плотность большого листа из такого материала.

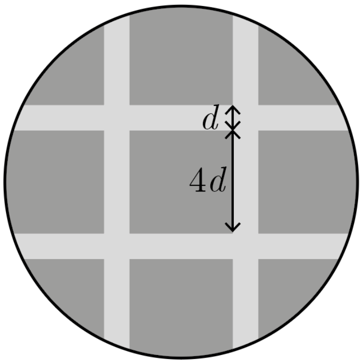

Рис. 2.3.19

Решение

По мере увеличения размеров листа материала роль его краев в общей плотности постепенно снижается, поэтому среднюю плотность большого листа следует вычислять как среднюю плотность одного элемента периодичности, границы которого изображены пунктиром на рис. 2.3.20. Объем V этого элемента равен $25dh$, где h — толщина листа материала.

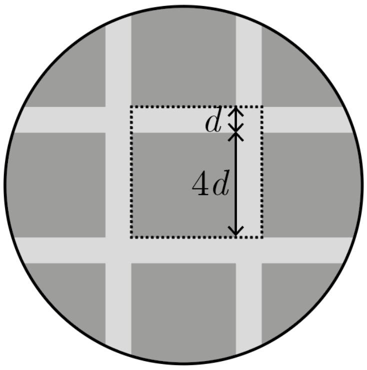

Рис. 2.3.20

Массу элемента найдем, сложив массы алюминия (индексы 1) и смолы (индексы 2)

$$m = m_1 + m_2 = \rho_1 V_1 + \rho_2 V_2 = \rho_1 9dh + \rho_2 16dh. \qquad (2.3.113)$$

Тогда искомая плотность окончательно равна

$$\rho = \frac{m}{V} = \frac{\rho_1 9\cancel{dh} + \rho_2 16\cancel{dh}}{25\cancel{dh}} \approx 1{,}74\,\text{г/см}^3. \qquad (2.3.114)$$

Погрешность $0{,}01\,\text{г/см}^3$.

Ответ: $(1{,}74 \pm 0{,}01)\,\text{г/см}^3$.

Задача 2.3.7.4. Домкрат (25 баллов)

Условие

На рис. 2.3.21 приведена схема устройства гидравлического домкрата. Его поршни представляют собой цилиндры с радиусами $R = 21$ см и $r = 3$ см. Чтобы поднимать при помощи этого домкрата груз $m = 1{,}4$ т, установленный на платформе большого цилиндра, к точке A рычага необходимо приложить силу не менее $F = 87{,}5$Н. Определите отношение $AB : BC$. Ускорение свободного падения $g = 9{,}8$ м/с^2. На рисунке точный масштаб не сохранен.

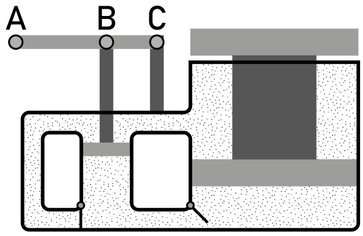

Рис. 2.3.21

Решение

Изображенный домкрат дает выигрыш в силе благодаря двум механизмам: рычагу и гидравлическому прессу. Выигрыш в силе, обеспечиваемый прессом, равен отношению площадей его цилиндров

$$\frac{mg}{F_1} = \frac{\cancel{\pi} R^2}{\cancel{\pi} r^2} \Rightarrow F_1 = mg\frac{r^2}{R^2}, \tag{2.3.115}$$

где F_1 — сила давления малого поршня.

В свою очередь, выигрыш в силе, обеспечиваемый рычагом, равен отношению его плеч, но так как рычаг закреплен в точке C, нужно сравнивать плечи AC и BC

$$\frac{F_1}{F} = \frac{AC}{BC} = \frac{AB + BC}{BC} = 1 + \frac{AB}{BC}. \tag{2.3.116}$$

Совместив эти уравнения, получим окончательно

$$\frac{AB}{BC} = \frac{F_1}{F} - 1 = \frac{mgr^2}{FR^2} - 1 = 2{,}2. \tag{2.3.117}$$

Погрешность 0,1.

Ответ: $2{,}2 \pm 0{,}1$.

Задача 2.3.7.5. Номинальная мощность (30 баллов)

Условие

Изучая электронагреватель прямого действия, ученик заметил, что увеличение подаваемой на него силы тока на $\Delta I = 0{,}1$ А над номинальным значением приводит к увеличению тепловой мощности, выделяемой прибором, на $\Delta P_1 = 44$ Вт, а уменьшение силы тока на ту же величину от номинальной, приводит к уменьшению мощности на $\Delta P_2 = 36$ Вт. Найдите номинальную мощность прибора, считая его сопротивление независимым от температуры.

Решение

Согласно закону Джоуля – Ленца, тепловая мощность электронагревателя равна

$$P = I^2 R, \tag{2.3.118}$$

где I — сила пропускаемого через него тока, а R — сопротивление прибора. Последнее по условиям задачи можно считать неизменным, поэтому, введя обозначения I_0 для номинальной силы тока и P_0 для номинальной мощности прибора, можно составить пропорции

$$\begin{cases} \dfrac{P_0 + \Delta P_1}{P_0} = \left(\dfrac{I_0 + \Delta I}{I_0}\right)^2, \\ \dfrac{P_0 - \Delta P_2}{P_0} = \left(\dfrac{I_0 - \Delta I}{I_0}\right)^2. \end{cases} \tag{2.3.119}$$

Обозначив $\frac{\Delta I}{I_0}$ буквой x и частично сократив дроби, приведем их к виду

$$\begin{cases} 1 + \dfrac{\Delta P_1}{P_0} = (1+x)^2, \\ 1 - \dfrac{\Delta P_2}{P_0} = (1-x)^2. \end{cases} \tag{2.3.120}$$

Эту систему можно решить, вычитая второе уравнение из первого

$$\frac{\Delta P_1 + \Delta P_2}{P_0} = 4x \Rightarrow x = \frac{\Delta P_1 + \Delta P_2}{4P_0} \tag{2.3.121}$$

и подставляя результат в любое уравнение системы (2.3.120)

$$\cancel{1} + \frac{\Delta P_1}{P_0} = \cancel{1} + \frac{\Delta P_1 + \Delta P_2}{2P_0} + \frac{(\Delta P_1 + \Delta P_2)^2}{16P_0^2}. \tag{2.3.122}$$

Домножим на $16P_0^2$

$$16\Delta P_1 P_0 = 8P_0(\Delta P_1 + \Delta P_2) + (\Delta P_1 + \Delta P_2)^2. \tag{2.3.123}$$

и выразим окончательно

$$P_0 = \frac{(\Delta P_1 + \Delta P_2)^2}{8(\Delta P_1 - \Delta P_2)} = 100\,\text{Вт}. \tag{2.3.124}$$

Погрешность 1 Вт.

Ответ: 100 ± 1 Вт.

2.3.8. Четвертая волна. Задачи 10–11 класса

Задачи четвертой волны предметного тура по по физике за 10–11 класс открыты для решения. Соревнование доступно на платформе Яндекс.Контест: https://contest.yandex.ru/contest/63483/enter/.

Задача 2.3.8.1. Шестеренки (10 баллов)

Условие

В сложной трансмиссии две шестеренки А и Б вращаются в различных частях механизма так, что угловая скорость вращения шестеренки А в 3 раза выше, чем шестеренки Б, но линейная скорость зубцов шестеренки Б в 2 раза выше, чем шестеренки А. Найдите отношение центростремительного ускорения зубцов шестеренки А к центростремительному ускорению зубцов шестеренки Б.

Решение

Два хорошо известных выражения для центростремительного ускорения a

$$a = \frac{v^2}{R} = \omega^2 R, \tag{2.3.125}$$

где R — радиус траектории, v — линейная скорость, ω — угловая скорость. Перемножив эти выражения, получим

$$a^2 = \frac{v^2}{\cancel{R}}\omega^2 \cancel{R} \Rightarrow a = v\omega. \tag{2.3.126}$$

Таким образом, центростремительное ускорение равно произведению линейной скорости на угловую, из чего следует

$$\frac{a_А}{a_Б} = \frac{v_А}{v_Б} \cdot \frac{\omega_А}{\omega_Б} = \frac{1}{2} \cdot \frac{3}{1} = 1{,}5. \tag{2.3.127}$$

Погрешность 0,01.

Ответ: $1{,}50 \pm 0{,}01$.

Задача 2.3.8.2. Маятник (15 баллов)

Условие

Заряженный металлический шарик закреплен на конце тонкой шелковой нити и несет заряд $q = 20$ мкКл. Другой конец нити закреплен к потолку. Определите массу шарика, если при помещении такого маятника в однородное электрическое поле, вектор напряженности которого направлен строго горизонтально и равен по модулю $E = 2{,}5$ кВ/м, сила натяжения нити после установления равновесия оказывается равна $T = 130$ мН. Ускорение свободного падения $g = 9{,}8$ м/с^2.

Решение

На шарик действуют три силы: горизонтально направленная сила электростатического взаимодействия $q\vec{E}$, вертикально вниз направленная сила тяжести $m\vec{g}$ и направленная вдоль нити сила ее натяжения $\vec{T}$. Разумеется, маятник может быть в равновесии, только если векторная сумма этих сил равна нулю, то есть эти три вектора образуют замкнутый треугольник

$$q\vec{E} + m\vec{g} + \vec{T} = \vec{0}. \tag{2.3.128}$$

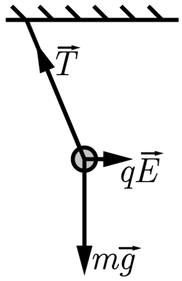

Рис. 2.3.22

Поскольку направления $\vec{E}$ и $\vec{g}$ известны, этот треугольник прямоугольный, а $\vec{T}$ — его гипотенуза. Тогда из теоремы Пифагора

$$q^2E^2 + m^2g^2 = T^2 \Rightarrow m = \frac{\sqrt{T^2 - q^2E^2}}{g} \approx 12{,}2\,\text{г}. \tag{2.3.129}$$

Погрешность $0{,}2$ г.

Ответ: $(12{,}2 \pm 0{,}2)$ г.

Задача 2.3.8.3. Автопилот (20 баллов)

Условие

Автомобиль с автопилотом запрограммирован таким образом, что при движении по прямой на трассе он всегда старается поддерживать дистанцию между собой и движущимся непосредственно перед ним автомобилем ровно такой же, как между собой и движущимся непосредственно за ним автомобилем. В некоторый момент движения скорости всех трех этих автомобилей были равны. Определите ускорение автомобиля, движущегося непосредственно за автопилотируемым, если модули ускорения самого автопилотируемого автомобиля и движущегося непосредственно перед ним в этот момент оба оказались равны $a = 0{,}2\,\text{м/с}^2$, но дистанция между ними при этом начала сокращаться. Колеса автомобилей движутся без проскальзывания, и автопилоту удается соблюдать требования своей программы.

Решение

Программа автомобиля означает, что (до тех пор, пока это позволяет мощность двигателя и сцепление колес) координата x вдоль оси, совпадающей с дорогой, автомобиля с автопилотом равна среднему арифметическому координат x_1 идущего впереди и x_2 идущего позади

$$x = \frac{x_1 + x_2}{2}. \tag{2.3.130}$$

Поскольку такая кинематическая связь справедлива в любые два момента времени t_1 и t_2, для средней скорости v автомобиля на автопилоте в проекции на $0x$ на любом промежутке времени можно записать

$$v_x = \frac{x(t_2) - x(t_1)}{t_2 - t_1} = \frac{x_1(t_2) + x_2(t_2) - x_1(t_1) - x_2(t_1)}{2(t_2 - t_1)} = \frac{v_{x1} + v_{x2}}{2}, \tag{2.3.131}$$

где использована та же система индексов. Таким образом, средняя скорость на любом промежутке времени и, следовательно, мгновенная скорость в любой момент времени автомобиля на автопилоте в проекции на $0x$ равна среднему арифметическому мгновенной скорости в этой же проекции впереди и позади идущих автомобилей. Повторение этих рассуждений приводит к аналогичному результату для ускорений

$$a_x = \frac{a_{x1} + a_{x2}}{2}. \tag{2.3.132}$$

Выразим из этого уравнения проекцию на $0x$ искомого ускорения замыкающего автомобиля a_{x2}

$$a_{x2} = 2a_x - a_{x1}. \tag{2.3.133}$$

По условиям задачи в рассматриваемый момент скорости всех трех автомобилей равны, а ускорения a_1 и a совпадают по модулю, но дистанция начинает сокращаться. Это возможно только если передний автомобиль тормозит — имеет отрицательную проекцию ускорения на направление движения, а автопилотируемый автомобиль, напротив, ускоряется (имеет положительную проекцию). Тогда напрямую из (2.3.133) получим

$$a_{x2} = 2a_x - a_{x1} = 2a - (-a) = 3a = 0{,}6\,\text{м/с}^2. \tag{2.3.134}$$

Погрешность $0{,}01\,\text{м/с}^2$.

Ответ: $(0{,}60 \pm 0{,}01)\,\text{м/с}^2$.

Задача 2.3.8.4. Лифт (25 баллов)

Условие

В лифте, движущемся вверх с некоторым ускорением a, сонаправленным его скорости, уронили без начальной скорости относительно лифта мячик с высоты $h_1 = 0{,}6\,\text{м}$ над уровнем пола лифта. Мячик абсолютно упруго ударился о пол, но своим ударом спровоцировал срабатывание системы аварийной остановки, в результате чего в момент удара ускорение лифта резко поменяло направление на противоположное, а его модуль возрос втрое. После отскока мячик поднялся до высоты $h_2 = 1{,}8\,\text{м}$. Определите a. Ускорение свободного падения $g = 9{,}8\,\text{м/с}^2$.

Решение

В системе отсчета, связанной с лифтом, начальное ускорение мяча равно $g + a$. Проходя с этим ускорением расстояние h_1, мяч приобретает скорость (относительно лифта) v, которую легко вычислить из соотношения

$$\frac{v^2}{2} = (g + a)h_1. \tag{2.3.135}$$

В момент удара резко меняется ускорение, но не скорость лифта, поэтому значение v при абсолютно упругом ударе по модулю остается неизменным.

В процессе подъема мяч уже имеет относительно лифта ускорение $g - 3a$, что позволяет записать аналогично

$$\frac{v^2}{2} = (g - 3a)h_2. \tag{2.3.136}$$

Совмещая эти равенства, получим окончательно

$$(g + a)h_1 = (g - 3a)h_2 \Rightarrow a(h_1 + 3h_2) = g(h_2 - h_1) \Rightarrow a = g\frac{h_2 - h_1}{h_1 + 3h_2} = 1{,}96\,\text{м/с}^2. \tag{2.3.137}$$

Погрешность $0{,}02$ м/с^2.

Ответ: $(1{,}96 \pm 0{,}02)$ м/с^2.

Задача 2.3.8.5. Эффект Лейденфроста (30 баллов)

Условие

Капля воды при температуре немного ниже температуры кипения упала на раскаленную поверхность, в результате чего $\alpha = 10^{-5}$ ее массы практически мгновенно испарилось. Известно, что $\eta = 3\%$ полученной каплей энергии пошло на работу расширяющегося пара над оставшейся частью капли.

На какую высоту «подпрыгнет» капля вертикально вверх в результате такого испарения, если сопротивлением воздуха ее движению, а также потерями тепла в окружающую среду и работой пара против воздуха можно пренебречь?

Удельная теплота парообразования воды равна $L = 2{,}26$ МДж/кг, ускорение свободного падения $g = 9{,}8\,\text{м/с}^2$.

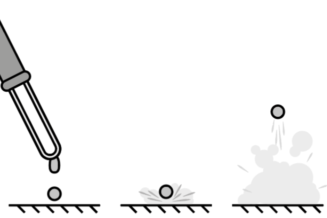

Рис. 2.3.23

Решение

Согласно первому началу термодинамики та теплота, полученная каплей, которая не пошла на работу расширяющегося пара над жидкой частью капли, ушла на изменение ее внутренней энергии; в данном случае — на испарение. Таким образом, можно записать для для этой части энергии

$$\Delta U = (1-\eta)Q = Lm\alpha, \tag{2.3.138}$$

где m — масса капли до испарения, а Q — общая полученная каплей теплота. Отсюда легко найти Q

$$Q = \frac{Lm\alpha}{1-\eta}. \tag{2.3.139}$$

В то же время работа пара над каплей полностью идет на увеличение ее механической энергии, которая в верхней точке траектории чисто потенциальна

$$A = \eta Q = (1-\alpha)mgh. \tag{2.3.140}$$

Выражая из этого уравнения h и подставляя в него Q, получаем

$$h = \frac{\eta Q}{(1-\alpha)mg} = \frac{L\alpha\eta}{(1-\alpha)(1-\eta)g} \approx 7{,}1\,\text{см}. \tag{2.3.141}$$

Разумеется, если величину $1-\alpha$ в этом выражении считать просто единицей, ответ не изменится в пределах любой разумной погрешности.

Погрешность $0{,}5\,$см.

Ответ: $(7{,}1 \pm 0{,}5)$ см.

2.4. Инженерный тур

Задачи первого этапа инженерного тура открыты для решения. Соревнование доступно на платформе Яндекс.Контест: https://contest.yandex.ru/contest/66704/enter/.

Задача 2.4.1. (25 баллов)

Темы: турбина, вращение, импульсы.

Условие

Измерение частоты оборотов паровой турбины происходит следующим образом: на роторе турбины установлена шестерня с определенным количеством зубьев, напротив шестерни установлен датчик Холла, фиксирующий магнитное поле и его напряженность. Каждый зуб, проходя мимо датчика, меняет магнитное поле, что влечет изменение напряжения на датчике. Датчик преобразует частоту оборотов в электрические прямоугольные импульсы. По количеству этих импульсов можно определить частоту вращения турбины. Для этого применяется электронное устройство — частотомер. Назовите главный компонент (микросхему), который используется в электрической схеме частотомера.

Ответ: счетчик.

Задача 2.4.2. (25 баллов)

Тема: программирование.

Условие

Управление скоростью вращения турбины осуществляется с помощью электромагнитного клапана, у которого есть две команды: «открыть» и «закрыть». Им соответствуют высокий (1) и низкий (0) логические уровни электрического сигнала соответственно. Необходимо разработать программу, автоматизирующую управление данным клапаном для защиты турбины от превышения частоты вращения. В нормальной эксплуатации (НЭ) значение частоты должно быть не выше 1 650 об/мин, и клапан должен быть открыт. При превышении данного значения клапан необходимо закрыть. Следует также учесть, что у турбины есть сервисный режим (СР) для проведения испытания механических бойков, при котором пороговое значение частоты вращения поднимается до 1685 об/мин. Условные обозначения: НЭ — 0, СР — 1.

Программу следует разрабатывать на языке C.

Программа должна принимать ТОЛЬКО целочисленные аргументы типа `int`.

На выходе программы необходимо получить цифру в формате `int`: «1» либо «0», что соответствует открытию или закрытию клапана.

Решение

Ниже представлено решение на языке C.

C

```c
#define OPEN 1
#define CLOSE 0
#define NORMAL_MODE 0
#define NORMAL_MODE_LIMIT 1650
#define SERVICE_MODE_LIMIT 1685

int getValveState(int frecuency, int mode) // функция, которая принимает значение частоты (frecuency) и режима (mode). Возвращает состояние открыть или закрыть
{
    if(mode == NORMAL_MODE) // если мы попали в нормальный режим
    {
        if(frecuency > NORMAL_MODE_LIMIT) // если значение частоты больше предела, нужно закрыть клапан, иначе открыть его.
        {
            return CLOSE;
        }
        return OPEN;
    }
    else // если попали в сервисный режим
    {
        if(frecuency > SERVICE_MODE_LIMIT) // если значение частоты больше предела, нужно закрыть клапан, иначе открыть его.
        {
            return CLOSE;
        }
        return OPEN;
    }
}
```

Тесты

Таблица 2.4.1

№ теста	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Аргумент 1	500	500	725	725	950	950	1400	1400	1650	1650
Аргумент 2	0	1	0	1	0	1	0	1	0	1
Результат	1	1	1	1	1	1	1	1	1	1
№ теста	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**
Аргумент 1	1660	1660	1685	1685	1850	1850	2075	2075	2300	2300
Аргумент 2	0	1	0	1	0	1	0	1	0	1
Результат	0	1	0	1	0	0	0	0	0	0

Задача 2.4.3. Вращение лопасти (50 баллов)

Темы: кинематика вращательного движения, ускорение при криволинейном движении.

Условие

Лопасть вращается со скоростью 1 400 об/мин, а ее радиус составляет 60 см.

1. Определите полное ускорение точки. Ответ дайте с точностью до целых в метрах в секунду за секунду.
2. Используя центр окружности в качестве начала координат, определите, по какому закону меняется значение координаты X со временем (какая функция описывает зависимость $x(t)$). Достаточно указать только вид зависимости без аргументов.

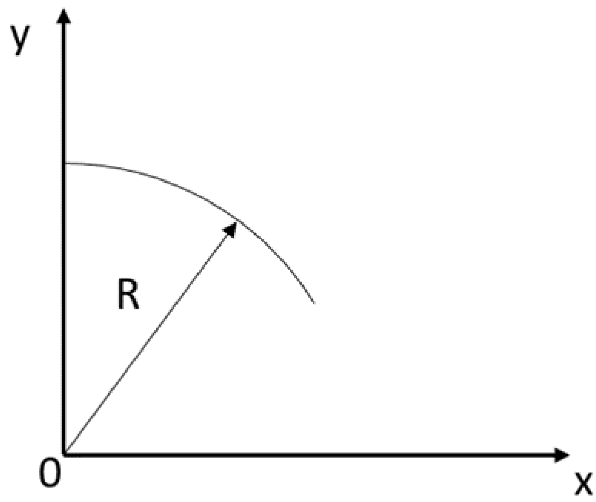

Рис. 2.4.1

3. Турбина начинает вращаться из неподвижного положения и развивает скорость 1 400 об/мин за 120 с. Определите полное ускорение точки в момент $t = 100$ с. Ответ дайте с точностью до целых в метрах в секунду за секунду.
4. Используя центр окружности в качестве начала координат, определите угол между вектором скорости и положительным направлением оси X в момент времени $t = 100$ с. Ответ дайте с точностью до целых в градусах с учетом знака.

Решение

1. Лопасть вращается с постоянной скоростью — вектор скорости в каждый момент времени изменяет только направление, сохраняя свое значение по модулю. Полное ускорение материальной точки будет складываться только из нормального (центростремительного) ускорения, которое можно найти по формуле:

$$a = \frac{v^2}{R}. \tag{2.4.1}$$

Переведем единицы измерения из оборотов в минуту:

$$1400\,\text{об/мин} = \frac{1400 \cdot 360 \cdot \pi}{180 \cdot 60}\,\text{рад/с} = 146{,}6\,\text{рад/с}.$$

Чтобы получить линейную скорость, умножим угловую скорость на радиус:

$$v = \omega R = 146{,}6 \cdot 0{,}6 = 87{,}9\,\text{м/с}.$$

Подставим полученное значение в формулу для вычисления нормального ускорения:

$$a_n = \frac{(87{,}9)^2}{0{,}6} = 12887\,\text{м/с}^2.$$

2. Вектор скорости точки в любой момент времени направлен по касательной к траектории движения. Отобразим этот вектор и спроецируем на заданные оси.

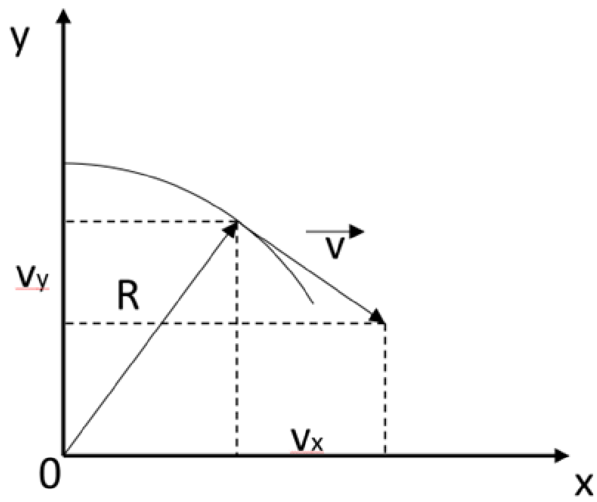

Рис. 2.4.2

V_x можно выразить через один из двух углов прямоугольного треугольника. Так как скорость и координата однозначно связаны, зависимость будет одна и та же с точностью до множителя перед тригонометрической функцией. Ответ: синус/косинус, sin/cos.

3. В данном случае, в отличие от п.1, вектор скорости изменяется не только по направлению, но и по модулю, и полное ускорение складывается из двух составляющих — тангенциального и нормального. Решение для поиска нормального ускорения приведено в п.1. Тангенциальную составляющую ускорения можно найти по формуле:

$$a_\tau = \frac{\Delta v}{\Delta t} = \frac{87{,}9}{120} = 0{,}73\,\text{м/с}^2.$$

Скорость в момент $t = 100$ с найдем, умножив тангенциальную составляющую ускорения на время:

$$v(100) = a_\tau t = 73\,\text{м/с}.$$

По формуле (2.4.1) найдем нормальную составляющую ускорения:

$$a_n = \frac{73^2}{0{,}6} = 8882\,\text{м/с}^2.$$

Модуль полного ускорения точки можно получить по теореме Пифагора:

$$a = \sqrt{a_n^2 + a_\tau^2} = \sqrt{8882^2 + 0{,}73^2} = 8882\,\text{м/с}^2.$$

4. Закон движения точки по окружности можно записать следующим образом:

$$x(t) = R\cos(\omega t),$$

$$y(t) = R\sin(\omega t).$$

Получим скорость, дифференцируя по t:

$$v_x(t) = -R\omega\sin(\omega t),$$

$$v_y(t) = R\omega\cos(\omega t).$$

Подставляя условия и результаты из предыдущих пунктов, получим:

$$v_x(100) = -0{,}6 \cdot 146{,}6\sin(146{,}6 \cdot 100) = -85{,}4\,\text{м/с},$$

$$v_y(100) = 0{,}6 \cdot 146{,}6\cos(146{,}6 \cdot 100) = 21{,}1\,\text{м/с}.$$

Определим тангенс угла и через него значение угла в градусах:

$$\text{tg}(\varphi) = \frac{v_x}{v_y} = -4{,}05,$$

$$\varphi = \arctan(-4{,}05) = -76^\circ.$$

Также принимается ответ 104°.

Ответ:

1. $12887 \pm$ 100 м/с^2;
2. любой из ответов: синус, косинус, sin, cos;
3. $8882 \pm$ 100 м/с^2;
4. -76° или 104°.

3. Второй отборочный этап

3.1. Работа наставника НТО на этапе

На втором отборочном этапе НТО участникам предстоит решать как индивидуальные, так и командные задачи в рамках выбранного профиля. Подготовка к этому этапу требует от них не только глубокого понимания предметной области, но и умения работать в команде, эффективно распределять роли и применять полученные знания на практике. Наставник играет здесь важную роль — он помогает участникам выстроить осмысленную и целенаправленную траекторию подготовки.

Вот основные направления, в которых наставник может поддержать участника:

- **Подготовка по образовательным программам НТО.** Наставник может готовить участников, используя готовые образовательные программы по технологическим направлениям, рекомендованные организаторами, а также адаптировать их под уровень подготовки школьников.
- **Разбор заданий прошлых лет.** Изучение задач второго отборочного этапа прошлых лет помогает участникам понять формат заданий, определить типовые ошибки и выработать стратегии решения.
- **Онлайн-курсы.** Участники могут пройти курсы по разбору задач прошлых лет или курсы, рекомендованные разработчиками отдельных профилей. Наставник может включить эти курсы в план подготовки, а также сопровождать процесс изучения и помогать с возникшими вопросами.
- **Анализ материалов профиля.** Совместный разбор методических материалов, размещенных на страницах профилей, помогает уточнить требования к участникам и направить подготовку на ключевые темы.
- **Практикумы.** Это важный элемент подготовки, позволяющий применять знания на практике. Наставник может:
  - ◇ организовать практикумы по методическим материалам с сайта профиля;
  - ◇ декомпозировать задачи заключительного этапа прошлых лет на отдельные элементы и проработать их с участниками;
  - ◇ провести анализ требуемых профессиональных компетенций и спланировать занятия для развития наиболее значимых из них;
  - ◇ направить участников на практикумы и мероприятия от организаторов, которые анонсируются в официальных сообществах НТО, например, в телеграм-канале для наставников: https://t.me/kruzhok_association.
- **Командная работа.** Одной из ключевых задач наставника на втором этапе является помощь в формировании команды или в поиске подходящей. Наставник может помочь участникам определить их сильные стороны, выбрать роль в команде и сориентироваться в процессе командообразования, включая участие в бирже команд в рамках конкретного профиля.

Если участники не прошли отборочный этап

Случается, что несмотря на усилия и серьезную подготовку, участники не проходят во второй или заключительный этап Олимпиады. В такой ситуации особенно важна поддержка наставника.

- **Поддержка и признание усилий.** Наставнику важно подчеркнуть ценность пройденного пути: полученные знания, навыки, преодоленные трудности и личностный рост. Это помогает участникам сохранить мотивацию и не воспринимать результат как окончательное поражение.
- **Рефлексия.** Полезно организовать встречу для обсуждения впечатления от участия, трудности, с которыми столкнулись школьники и то, что они узнали о себе и команде. Наставник может направить разговор в конструктивное русло: какие выводы можно сделать? Что сработало хорошо? Что можно улучшить?
- **Анализ ошибок и пробелов.** Наставник вместе с участниками анализирует, какие темы вызвали наибольшие затруднения, чего не хватило в подготовке — теоретических знаний, практических навыков, командного взаимодействия. Это позволяет выстроить более эффективную стратегию на будущее.
- **Планирование дальнейшего пути.** Участникам можно предложить:
    - ◇ продолжить углубленное изучение профиля или смежных направлений;
    - ◇ заняться проектной деятельностью, которая укрепит знания и навыки;
    - ◇ сформировать план по подготовке к следующему циклу НТО, начиная с работы над типовыми заданиями и курсами.
- **Создание устойчивой мотивации.** Важно показать школьникам, что участие в НТО — это не просто соревнование, а часть большого образовательного маршрута. Даже неудачный результат может стать толчком к профессиональному росту, если воспринимать его как точку развития, а не как конец пути.

Таким образом, наставник помогает участникам не только готовиться к этапам НТО, но и справляться с неудачами, выстраивать долгосрочную стратегию и сохранять интерес к инженерному и технологическому творчеству.

3.2. Инженерный тур

Задачи отборочного этапа непосредственно связаны с заключительной инженерной задачей и позволяют участникам наработать компетенции, необходимые на заключительном этапе, а также навыки командного взаимодействия.

3.2.1. Индивидуальные задачи

Индивидуальные задачи второго этапа инженерного тура открыты для решения. Соревнование доступно на платформе Яндекс.Контест: https://contest.yandex.ru/contest/69910/enter/.

Задача 3.2.1.1. Задача по математике (25 баллов)

Тема: численное решение дифференциальных уравнений.

Условие

Паровая турбина, изначально находившаяся в состоянии покоя, начинает раскручиваться под действием пара с постоянным давлением. Определите время в секундах, спустя которое скорость вращения турбины достигнет уровня, равного 95% от номинального (установившегося), если известно, что процесс разгона вращения турбины описывается следующим дифференциальным уравнением:

$$10\frac{df}{dt} + f(t) = 1\,500,$$

где f — скорость вращения турбины, об/с.

Ответ дайте с точностью до десятых долей.

Примечания

Для решения задачи рекомендуется применить построение разностной схемы методом Эйлера, которую, при необходимости, можно легко запрограммировать.

Решение

Метод Эйлера является простейшим алгоритмом численного решения обыкновенных дифференциальных уравнений.

Пусть имеется дифференциальное уравнение $\frac{dy}{dt} = f(y, t)$, где f — некоторая функция. Для численного решения этого уравнения методом Эйлера задается размер шага по времени Δt. При этом алгоритм вычисления значений функции $y(t)$ в моменты времени, определяемые размером шага Δt, имеет вид:

$$y_i = y_{i-1} + \Delta t \cdot f_{i-1}, \tag{3.2.1}$$

где i — номер шага (итерации). Другими словами, запись y_i означает значение функции y в момент времени, равный $i \cdot \Delta t$, т. е. $y_i = y(i \cdot \Delta t)$.

По условию задачи динамический процесс задан дифференциальным уравнением:

$$10\frac{df}{dt} + f(t) = 1\,500,$$

для которого можем записать:

$$\frac{df}{dt} = 150 - 0{,}1 \cdot f(t).$$

Разностная схема решения этого уравнения методом Эйлера в соответствии с (3.2.1) будет иметь следующий вид:

$$f_i = f_{i-1} + \Delta t \cdot (150 - 0{,}1 \cdot f_{i-1}).$$

Ответ: считать правильным ответ из диапазона [28,5; 30,5].

Задача 3.2.1.2. Задача по физике (25 баллов)

Тема: термодинамика.

Условие

В паровой турбине на вход подается пар высокой температуры под высоким давлением. По мере прохождения через турбину пар расширяется, чтобы сохранить скорость потока, и теряет температуру. Поток газа, прошедший через лопатки турбины, имеет более низкое давление и температуру. Чтобы эффективнее использовать оставшийся пар, применяют специальное устройство — конденсатор, в котором возможно восстановить термодинамические характеристики пара для дальнейшего использования.

Для оценки и расчета параметров проходящих в турбине процессов используют is-диаграмму (hs) водяного пара, в которой графически связаны между собой давление, температура, энтальпия и энтропия пара.

Диаграмма: `https://disk.yandex.ru/i/wRppOQdHGn6HIw`.

Пусть турбина работает с начальными параметрами $p_0 = 7$ МПа, $t_0 = 440$ °C и давлением пара в конденсаторе $p_k = 4$ кПа.

Определите эффективную мощность турбины, если расход пара $D = 6{,}3$ кг/с и относительный КПД турбины $\mu = 0{,}68$. Энтропию процесса считать постоянной. Ответ дайте в мегаваттах с точностью до десятых.

Подсказка

Оценить изменение энергии газа можно, вычисляя изменение энтальпии в процессе. Энтальпия (теплосодержание насыщенного газа) — количество теплоты, необходимое для повышения температуры 1 кг вещества от абсолютного нуля до пара заданной температуры. В общем случае энтальпия является функцией температуры и давления. Как и в случае с внутренней энергией, чаще надо знать изменение энтальпии.

При использовании диаграммы получаемые значения округляйте до ближайшего деления шкалы.

Решение

С помощью is-диаграммы оценим, как изменяется энтальпия в процессе.

Для начального состояния 7 МПа = 70 бар:

$$S_0 = 6{,}6\ \text{кДж/(кг} \cdot \text{К)};$$
$$I_0 = 3\,100\ \text{кДж/кг}.$$

Для конечного состояния (в конденсаторе) 4 кПа = 0,04 бар:

$$S_1 = 6{,}6\ \text{кДж/(кг} \cdot \text{К)};$$
$$I_1 = 2\,000\ \text{кДж/кг}.$$

Тогда изменение энтальпии в процессе:

$$I_0 - I_1 = 1\,100\ \text{кДж/кг}.$$

Эффективная мощность турбины:

$$P = D \cdot (I_0 - I_1) \cdot \mu = 6{,}3 \cdot 1\,100 \cdot 0{,}68 = 4\,712\ \text{кВт} = 4{,}7\ \text{МВт}.$$

Ответ: 4,7 МВт (принимаются ответы в диапазоне [4,5; 4,9] МВт).

Задача 3.2.1.3. Задача на программирование (25 баллов)

Тема: система дифференциальной защиты турбоустановки.

Условие

Дифференциальная защита для паровой турбины представляет собой критически важный элемент системы управления, обеспечивающий безопасность и надежность ее работы. Паровые турбины, как ключевые компоненты энергетических установок, подвергаются различным нагрузкам и потенциальным аварийным ситуациям. Поэтому внедрение эффективной системы дифференциальной защиты позволяет оперативно обнаруживать неисправности и предотвращать серьезные повреждения оборудования.

В данной задаче сосредоточимся на разработке программы работы дифференциальной защиты для системы регулирования турбины.

Дифференциатор предназначен для обеспечения защиты турбины при сбросах нагрузки и наличии положительной скорости изменения частоты вращения ротора турбины. Он включается при наличии ускорения вращения турбины и превышения максимально допустимой скорости ее вращения и действует в сторону уменьшения вращения. Выходной сигнал дифференциатора формируется в зависимости от значения скорости изменения частоты вращения. Зависимость представляет собой кусочно-линейную функцию.

Создание программного кода, реализующего данный алгоритм, позволит обеспечить надежный мониторинг и защиту турбины, гарантируя ее безопасное и эффективное функционирование в течение всего эксплуатационного периода.

Формулировка задачи

Разработать программный код для реализации алгоритма системы дифференциальной защиты паровой турбины в рамках системы регулирования. Блок-схема алгоритма изображена на рис. 3.2.1. Кусочно-линейная функция представлена на рис. 3.2.2.

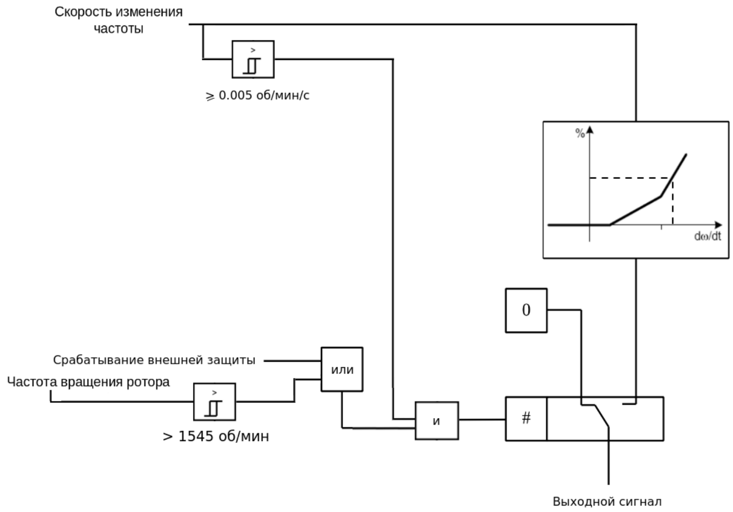

Рис. 3.2.1. Блок-схема алгоритма

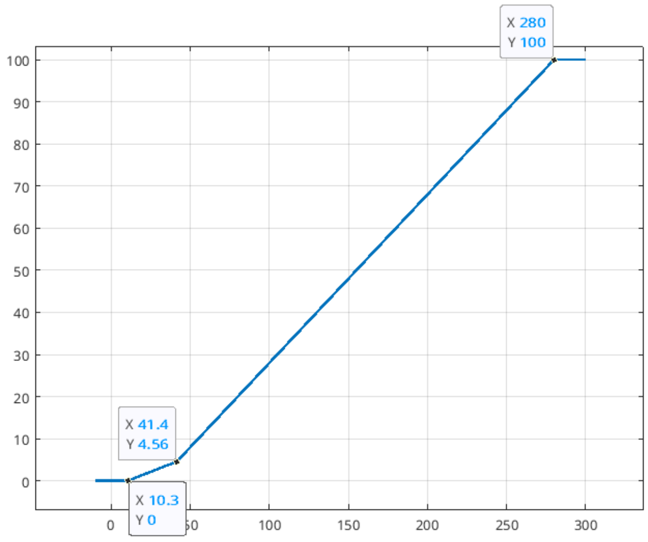

Рис. 3.2.2. График кусочно-линейной функции

Формат входных данных

На поток стандартного ввода программы подаются три числа: значение скорости изменения частоты вращения турбины, частота вращения и флаг срабатывания внешней защиты турбины (0 или 1).

Формат выходных данных

В поток стандартного вывода программы выводится одно число — аналоговое значение выходного сигнала алгоритма (округленное до трех знаков после запятой).

Примеры

Пример №1

Стандартный ввод
0 1205 0

Стандартный вывод
0.000

Пример №2

Стандартный ввод
0 1300 1

Стандартный вывод
0.000

Пример №3

Стандартный ввод
0.001 1600 0

Стандартный вывод
0.000

Пример №4

Стандартный ввод
5 1600 0

Стандартный вывод
0.000

Пример №5

Стандартный ввод
10.4 1567 0

Стандартный вывод
0.015

Пример №6

Стандартный ввод
42.189 1589 0

Стандартный вывод
4.876

Пример №7

Стандартный ввод
44.75 1566 0

Стандартный вывод
5.900

Пример №8

Стандартный ввод
281 1549 0

Стандартный вывод
100.000

Пример №9

Стандартный ввод
310.7 1549 0

Стандартный вывод
100.000

Пример №10

Стандартный ввод
19 1500 0

Стандартный вывод
0.000

Пример №11

Стандартный ввод
131.56 1700 0

Стандартный вывод
40.624

Пример №12

Стандартный ввод
133.05 1400 1

Стандартный вывод
41.220

Пример №13

Стандартный ввод
212 1534 1

Стандартный вывод
72.800

Требования к коду

Код должен быть структурированным, с использованием комментариев для улучшения читаемости, а также соответствовать стандартам надежности и безопасности программного обеспечения. Программу следует разрабатывать на языке C.

Решение

Ниже представлено решение на языке C.

C

```c
#include <stdbool.h>
#include <stdlib.h>
#include <stdio.h>

#define FREQ_CHANGE_THRESHOLD 0.005 // Уставка по ускорению
#define FREQ_THRESHOLD 1545 // Уставка по частоте

// Нелинейная функция
float interpolate(float x) {
    // Точки графика
    float x1=10.3, x2=41.4, x3=280;
    float y1=0, y2=4.56, y3=100;
    // Коэффициенты
    float k1=(y2-y1)/(x2-x1);
    float b1=y1-(x1*k1);
    float k2=(y3-y2)/(x3-x2);
    float b2=y3-(x3*k2);
    if (x <= x1) return 0;
    if (x < x2) return k1*x + b1;
    if (x < 280) return k2*x + b2;
    return 100;
}

int main(int argc, const char** argv) {
    float freqChange; // Значение ускорения ротора (об/мин/с)
    scanf("%f", &freqChange);
    int freqRotor; // Значение частоты ротора (об/мин)
    scanf("%d", &freqRotor);
    int protectionForce; // Срабатывание внешней системы защиты
    scanf("%d", &protectionForce);
    float y; // Выходной сигнал

    // Проверка входных условий
    bool freqChanged = freqChange >= FREQ_CHANGE_THRESHOLD; // Ускорение выше уставки
    bool rotorTooFast = freqRotor > FREQ_THRESHOLD; // Частота вращения выше уставки
    // Условия включения алгоритма диф. защиты:
    // - наличие ускорения ротора
    // - превышение граничного значения частоты вращения или срабатывания канала внешней защиты.
    bool allow = freqChanged && (protectionForce || rotorTooFast);

    // Если условия соблюдены, формируем выходной сигнал
    if (allow) {
        y = interpolate(freqChange);
    } else {
```

```
            // Условия не соблюдены
            y = 0;
        }
        printf("%.3f\n", y);
        return 0;
    }
```

Задача 3.2.1.4. Инженерная задача (25 баллов)

Тема: регуляторы турбины.

Условие

Необходимо перечислить все типы регуляторов (состоящие из одного, двух или трех звеньев), которые не входят в состав основных регуляторов электронной части системы регулирования турбины.

Текст легенды. Введение в регуляторы

Управление процессами и их параметрами в промышленности является главной областью применения регулирующих приборов и устройств. В основном они существуют в форме отдельных регулирующих приборов или программных регуляторов и применяются в машиностроении для координации и коррекции параметров температуры, давления, потока жидкости, количества, дозирования, уровня, положения, позиции, скорости, расстояния, а также в технологии производства для контроля и упорядочивания значений концентрации, проводимости, вязкости, плотности, химического состава и т. д.

Почти каждое регулирование определенной физической величины дает свои собственные специфические проблемы, которые зачастую определяются применяемыми исполнительными устройствами и датчиками.

Регуляторы — в смысле собственно регулирования — есть функциональные элементы, которые в зависимости от измеренной техническим датчиком величины процесса воздействуют по математически точному правилу на физическую величину в замкнутом контуре с помощью активного органа (рис. 3.2.3).

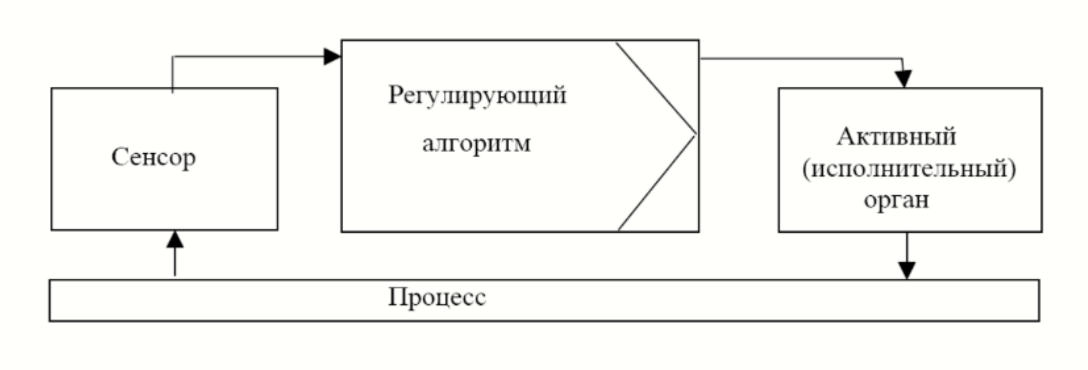

Рис. 3.2.3. Принцип прохождения сигнала при регулировании

Практически регуляторы содержат не только вычислительное правило (алгоритм), но имеют и ряд управляющих функций для обслуживания, наблюдения, обеспечения безопасности и возможностей переключений в контуре управления.

Все в дальнейшем описываемые регуляторы (PID-регуляторы, т. е. состоящие из звеньев пропорциональной передачи, интегрирующих и дифференцирующих), делятся по назначению:

- образование входного сигнала,
- формирование требуемого значения,
- формирование регулирующей разности,
- алгоритм (P, I, D),
- формирование выходного сигнала.

Для каждого звена регулятора есть ряд различных дополнительных — по выбору — функций, применение которых существенно влияет на характеристики регулятора и определяет классификацию множества регуляторов и их типов.

Общее описание систем турбины

Второй контур АЭС не является радиоактивным и состоит из парогенераторов, главных паропроводов, турбоагрегата, вспомогательного оборудования, обслуживающих систем, оборудования деаэрации, подогрева и подачи питательной воды в парогенераторы.

урбоустановка включает в себя паровую турбину и генератор, монтируемый на общем фундаменте с турбиной. Турбина снабжается конденсационным устройством, регенеративной установкой для подогрева питательной воды, сепараторами-пароперегревателями, а также имеет нерегулируемые отборы пара на подогреватели системы регенерации, на собственные нужды станции и на подогрев добавки химически очищенной воды в цикл.

Турбоустановка имеет возможность отпуска пара на бойлера теплосети из нерегулируемых отборов для работы на влажном паре, а также производит промежуточную сепарацию и промежуточный перегрев пара.

Турбина предназначена для непосредственного привода генератора переменного тока, что обеспечивает преобразование тепловой энергии в электрическую. Генератор должен иметь полное водяное охлаждение и допускать без ограничения во времени работу с максимальной мощностью.

Принципиальная тепловая схема турбоустановки включает в себя:

- четыре ступени подогревателей низкого давления (ПНД),
- деаэратор (Д),
- две ступени подогревателей высокого давления (ПВД).

Принятая схема предназначена для регенеративного подогрева основного конденсата (ОК) и питательной воды в режимах нормальной эксплуатации.

Пар из парогенератора через блоки стопорных и регулирующих клапанов поступает в цилиндры высокого давления (ЦВД) турбины.

Из камер отбора за ступенями каждого из потоков пар отбирается на регенеративный подогрев в подогреватели высокого давления и деаэратор. Из выхлопа ЦВД пар отбирается на ПНД. Основной расход пара после ЦВД направляется на сепарацию и промперегрев в сепараторе-пароперегревателе (СПП). После СПП пар через блоки клапанов низкого давления (последовательно установленные стопорные и регулирующие поворотные заслонки) поступает в цилиндры низкого давления (ЦНД). Из ЦНД осуществляется отбор пара на регенеративные ПНД.

На рис. 3.2.4 представлена принципиальная тепловая схема турбоустановки.

Система регулирования турбины выполняется электрогидравлической и состоит из электронной и гидравлической частей. Электронная часть системы регулирования турбины выполняется на современной микропроцессорной технике, применяемой для всей АЭС. Электронная часть системы защиты турбины обеспечивает закрытие всех паровпускных органов турбины с целью предотвращения аварийной ситуации.

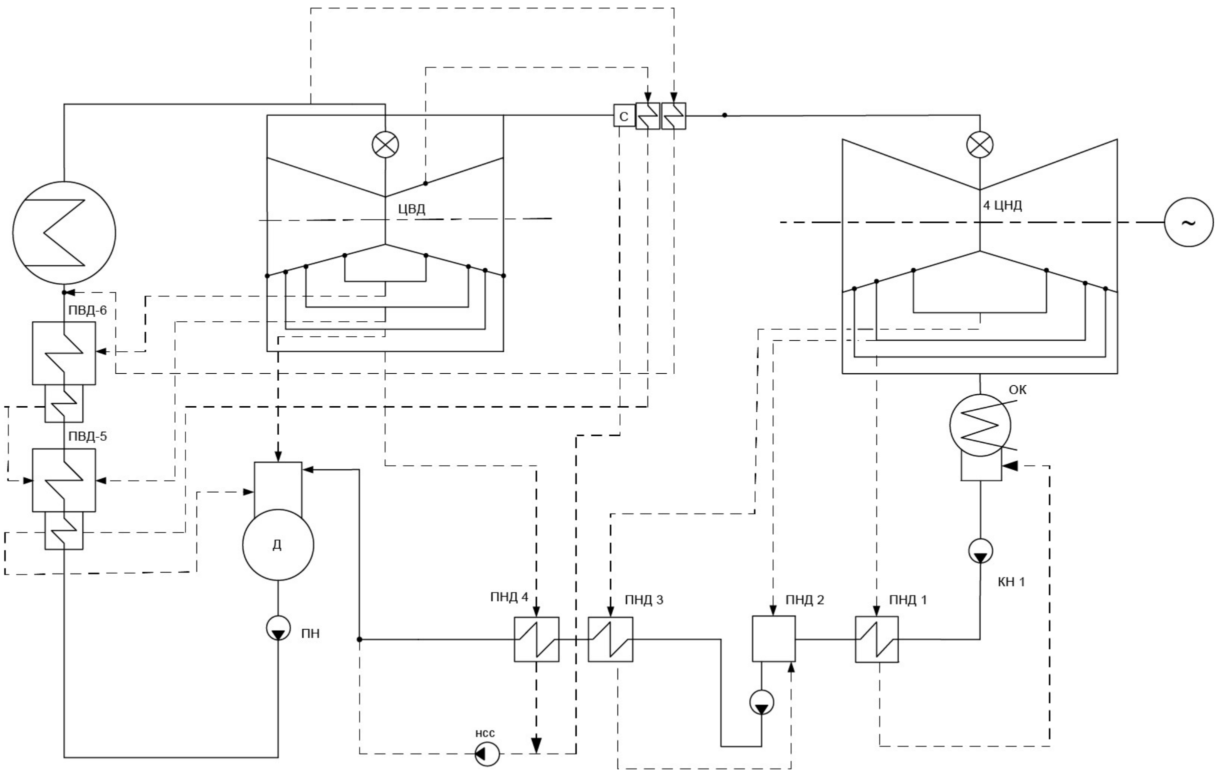

Рис. 3.2.4. Принципиальная тепловая схема турбоустановки

Ниже кратко перечислены и описаны основные регуляторы электронной части системы регулирования турбины в зависимости от используемого технологического параметра.

Температура

Регулятор температуры (РТ) пара после СПП — пропорционально-интегральный регулятор, обеспечивает поддержание температуры пара после СПП в соответствии с заданной величиной.

Задание по температуре пара формируется в соответствии с предустановленной зависимостью от величины электрической мощности генератора.

В качестве сигнала обратной связи регулятор получает результат обработки по заданному алгоритму сигналов от четырех измерений температуры пара после СПП.

Рассогласование, полученное как разница между заданной и текущей температурой, поступает на ПИ-регулятор температуры, выходной сигнал которого поступает в алгоритм для формирования необходимой выходной характеристики регулирующего клапана греющего пара.

Давление

Регулятор давления (РД) — пропорционально-интегральный регулятор, который обеспечивает поддержание давления пара в главном паровом коллекторе (ГПК) в соответствии с заданной величиной.

Задание по давлению исходно устанавливается равным номинальному, вместе с тем предусмотрена возможность коррекции задания оператором.

Регулятор включается:

- по команде оператора или АСУ ТП энергоблока;
- автоматически, при отключении систем регулирования давления пара в ГПК (регуляторы давления реактора и быстродействующей редукционной установки сброса пара в конденсатор (БРУ-К)), а также при срабатывании предупредительной защиты, обеспечивающей автоматическое снижение нейтронной мощности реактора.

Мощность

Регулятор мощности (РМ) — пропорционально-интегральный регулятор, который обеспечивает регулирование электрической нагрузки генератора в соответствии с заданной величиной в регулировочном диапазоне нагрузок.

Согласно концепции автоматического управления энергоблока АЭС с ВВЭР, всегда один из автоматических регуляторов энергоблока находится в режиме поддержания заданного давления пара в ГПК. Поэтому при работе генератора в сети турбинный регулятор может быть включен в режим РМ только при условии, что давление пара в ГПК поддерживается автоматическим регулятором мощности реактора (АРМ) или регулятором БРУ-К.

Сигнал задания для РМ формируется контроллером задания мощности системы управления, принимающим внешние входные сигналы:

- от оператора — конечное задание мощности, темп изменения мощности, максимальная и минимальная мощность;
- от центрального регулятора — задание мощности вторичного регулирования;
- от противоаварийной автоматики (ПАА) — задание мощности в послеаварийном режиме;
- от ПАА и технологической автоматики — команды автоматических разгрузок при возникновении ограничений на оборудовании.

Скорость

Регулятор скорости (РС) обеспечивает устойчивое регулирование частоты вращения ротора турбоагрегата в соответствии с заданием.

Структурно РС состоит из двух компонентов: неотключаемого статического регулятора и ПИ-регулятора в составе турбинного регулятора мощности/скорости и давления (ТРМД).

Статический РС всегда включен в работу. При отказах оборудования системы управления, приводящих к неработоспособности РС (включая недостоверность измерения частоты вращения ротора), формируется дискретный сигнал в систему защит на останов турбины.

Интегральная составляющая РС включается в работу в режиме регулирования скорости ТРМД посредством подключения на вход ПИ-регулятора сигнала рассогласования по частоте вращения. Режим регулирования скорости ТРМД автоматически активируется, если генератор не включен в сеть и обеспечивает управление турбиной:

- при развороте турбоагрегата;
- при работе генератора на холостом ходу;
- при синхронизации генератора с сетью;
- при работе генератора на изолированную от сети нагрузку.

В режиме регулирования скорости ТРМД для оператора или для внешней автоматики недоступно разомкнутое управление органами парораспределения турбины. Реализация управления возможна только посредством изменения задания регулятору скорости.

Положение

Регулятор положения (РП) регулирующих клапанов турбины (регулятор ручного управления парораспределением турбины) обеспечивает поддержание выходного сигнала ТРМД на заданном уровне. При изменении задания ПИ-регулятор ТРМД обеспечивает соответствующее изменение выходного сигнала с предустановленным для конкретного режима работы темпом.

Сигнал рассогласования проходит через селектор минимума с сигналами рассогласования ограничительных регуляторов давления и, в зависимости от знака, формирует сигналы управления ПИ-регулятором ТРМД. ПИ-регулятор работает в режиме интегрирующего задатчика так, чтобы его выходной сигнал изменялся к заданному значению с постоянной скоростью.

Режим ручного управления парораспределением турбины может быть активирован только при работе генератора в сети в следующих случаях:

- автоматически, по сигналу о включении генератора в сеть, обеспечивая взятие начальной нагрузки генератора подачей задания на задатчик РП;
- автоматически, по сигналам технологической и противоаварийной автоматики, для разгрузки генератора в разомкнутом контуре управления до заданной мощности с заданной скоростью;
- автоматически, при блокировке в системе управления режима регулирования мощности или режима регулирования давления пара в ГПК по причине недостоверности измерения соответствующего параметра;
- по команде оператора.

Описание основного функционального блока регуляторов в SimInTech

Блок ПИД-регулятора реализует математическую модель регуляторов, состоящих из одного звена: П-, И-, Д-регуляторы, двух звеньев: ПИ-, ПД-регуляторы или трех звеньев: ПИД-регулятор, в зависимости от заданных свойств.

Блок реализует передаточную функцию ПИД-регулятора следующего вида:

$$W(s) = K_p + \frac{K_i}{s} + \frac{K_d \cdot s}{T_d \cdot s + 1},$$

где K_p, K_i, K_d — коэффициенты усиления пропорциональной, интегрирующей и дифференцирующей составляющих регулятора соответственно, T_d — постоянная времени дифференцирования.

ПИД-регулятором называется устройство, применяемое в контурах управления, оснащенных звеном обратной связи (рис. 3.2.5).

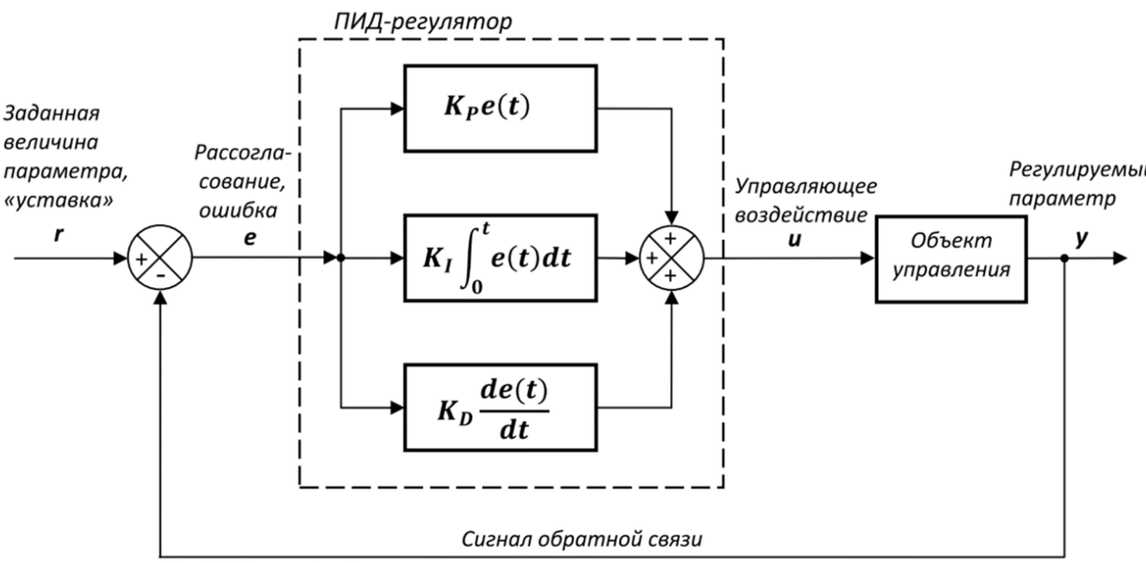

Рис. 3.2.5. Структурная схема системы управления с обратной связью с ПИД-регулятором

Выходной сигнал ПИД-регулятора определяется тремя слагаемыми: первое слагаемое пропорционально величине сигнала рассогласования (ошибки), второе — интегралу сигнала рассогласования, третье — его производной:

$$u(t) = P + I + D = K_p \cdot e(t) + K_i \cdot \int_0^t e(\tau)d\tau + K_d T_d \cdot \frac{de}{dt}.$$

Если какой-то из трех компонентов не используется, то регулятор будет пропорциональным, интегрирующим, дифференцирующим, или пропорционально-дифференцирующим, или пропорционально-интегрирующим, соответственно используемым компонентам.

Входы

- **in** — сигнал ошибки.

Выходы

- **out** — управляющее воздействие.

Свойства:

- **Тип регулятора** — режим работы регулятора может принимать следующие значения:
    - ◇ `P` — пропорциональный регулятор;
    - ◇ `I` — интегрирующий регулятор;
    - ◇ `D` — дифференцирующий регулятор;
    - ◇ `PI` — пропорционально-интегрирующий регулятор;
    - ◇ `PD` — пропорционально-дифференцирующий;
    - ◇ `PID` — пропорционально-интегрально-дифференцирующий регулятор.
- **Начальные условия** — вектор начальных значений $y_i(0)$ выходной величины блока.
- **Пропорциональная составляющая** — K_p — значение коэффициента усиления П-составляющей, появляется в режимах работы `P`, `PI`, `PD`, `PID`.
- **Интегральная составляющая** — K_i — значение коэффициента усиления И-составляющей, появляется в режимах работы `I`, `PI`, `PID`.
- **Дифференциальная составляющая** — K_d — значение коэффициента усиления Д-составляющей, появляется в режимах работы `D`, `PD`, `PID`.
- **Постоянная времени дифференцирования**, с — T_d — появляется в режимах работы `D`, `PD`, `PID`.

Решение

Внимательно ознакомившись с легендой задачи, участник должен выбрать из текста легенды правильный ответ.

Ответ:

Состоящих из одного звена: П-, И-, Д-регуляторы.
Состоящих из двух звеньев: ПД-регуляторы.
Состоящих из трех звеньев: ПИД-регулятор.

3.2.2. Командная задача

Командная задача второго этапа инженерного тура открыта для решения. Соревнование доступно на платформе Яндекс.Контест: https://contest.yandex.ru/contest/69933/enter/.

Задача 3.2.2.1. Регулятор давления пара в главном паровом коллекторе (ГПК) турбоустановки с реактором типа ВВЭР (100 баллов)

Условие

Регулятор давления паровой турбины представляет собой пропорционально-интегральный регулятор, который предназначен для поддержания давления пара в главном паровом коллекторе турбины в соответствии с заданной величиной.

По команде оператора АЭС регулятор давления пара в ГПК безударно включается в работу и выводит давление пара на нижнюю границу допустимого рабочего диапазона изменения давления. Рабочий допустимый диапазон изменения составляет $\pm 5\%$ от номинального значения давления пара ($p_{\text{ном}} = 6{,}8$ МПа). При этом на вход регулятора давления поступает нулевое рассогласование, сформированное в виде разности между заданным значением и равным ему по величине текущим значением давления пара в ГПК.

Начальное состояние системы:

- регулятор реактора находится в режиме поддержания постоянного тепловыделения в парогенераторе,
- активен турбинный регулятор давления пара в ГПК,
- заданное начальное значение давления пара в ГПК соответствует нижней границе допустимого рабочего диапазона изменения давления,
- показания датчика давления не отличаются от заданного значения,
- выходное управляющее воздействие регулятора отсутствует.

Через некоторое время оператор устанавливает величину конечного задания давления, равной номинальному значению давления ($p_y = p_{\text{ном}}$). С этого момента на вход регулятора давления начинает поступать ненулевое рассогласование, сформированное в виде разности между новым заданным значением и текущим значением давления пара в ГПК.

Уравнение регулятора давления пара в ГПК можно представить в виде:

$$(p_y - p) \cdot K_p + K_i \int_0^t (p_y - p) dt = u_p,$$

где

- p — давление пара в ГПК по показаниям датчика,
- p_y — заданное оператором давление пара в ГПК,
- K_p — коэффициент пропорциональности ($K_p = 1$),
- K_i — коэффициент интегрирования ($K_i = 2$),
- u_p — управляющее воздействие.

Уравнение объекта управления, состоящего из оборудования первого парового отсека турбины, можно представить в упрощенном виде:

$$T_p \frac{dp}{dt} + p = u_p,$$

где T_p — постоянная времени первого парового отсека ($T_p = 0{,}9$).

Обратная связь системы управления содержит датчик давления пара, динамика которого описывается линейным уравнением с запаздыванием:

$$y_i(t) = x_i(t - T_d),$$

где T_d — время задержки сигнала датчика давления ($T_d = 0{,}05$ с).

Рис. 3.2.6. Общая структурная схема системы регулирования давления пара в ГПК

На рис. 3.2.6 представлена общая структурная схема математической модели системы с замкнутым контуром регулирования давления свежего пара для исследования переходного процесса при ступенчатых изменениях задания по давлению пара в ГПК.

С помощью указанных выше параметров системы посредством функционально-блочного программирования платформы SimInTech произведите расчет времени установления системой управления номинального давления пара в ГПК.

SimInTech — это мощная среда для моделирования, анализа и проектирования сложных динамических систем. Платформа предназначена для инженеров, исследователей и разработчиков, которые работают с математическими моделями, включая механические, электрические, гидравлические, тепловые и другие системы. SimInTech позволяет создавать виртуальные прототипы, проводить симуляции, оптимизировать параметры и тестировать системы в различных условиях.

Основные возможности SimInTech:

- гибкое моделирование с использованием блок-схем и математических моделей;
- поддержка сложных технических систем;
- интеграция с другими инструментами и языками программирования;
- визуализация и анализ результатов моделирования;
- возможность создания пользовательских библиотек и компонентов.

SimInTech активно используется в таких областях, как аэрокосмическая промышленность, энергетика, робототехника, автомобилестроение и многих других.

Дополнительная информация о платформе и бесплатная версия размещены на официальном сайте: https://simintech.ru.

В ответе укажите максимальное значение давления (с точностью до 0,01 МПа), полученное в процессе регулирования, определите, через какое время (с точностью до 0,1 с) после изменения задания оператором давление в ГПК достигнет впервые номинального значения 6,8 МПа, а также найдите, через какое время значение вернется в трубку установившегося номинального значения 6,8 ± 0,01 МПа.

Решение

Схема SimInTech представлена на рис. 3.2.7.

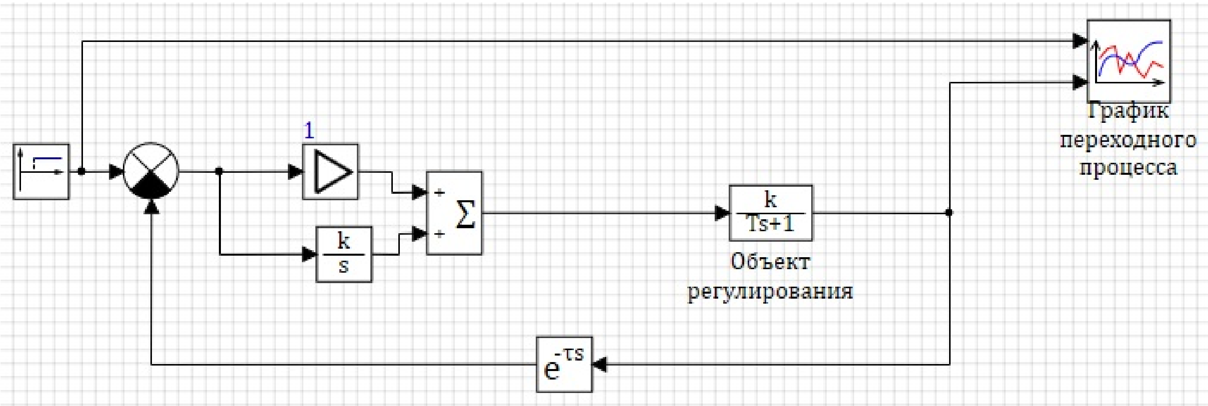

Рис. 3.2.7. Схема SimInTech

Основные настройки функциональных блоков схемы показаны ниже:

1. Настройка блока «Ступенька» библиотеки «Источники», рис. 3.2.8.

Свойства | Общие | Порты | Визуальные слои

Название	Имя	Формула	Значение
Время срабатывания	t	1	[1]
Начальное состояние	y0	6.46	[6.46]
Конечное состояние	yk	6.8	[6.8]

Рис. 3.2.8

2. Настройка блока «Усилитель» библиотеки «Операторы», рис. 3.2.9.

Свойства | Общие | Порты | Визуальные слои

Название	Имя	Формула	Значение
Коэффициент усиления	a		1
Отображать формулу	formula_visible		☐ Нет

Рис. 3.2.9

3. Настройка блока «Интегратор» библиотеки «Динамические», рис. 3.2.10.

Свойства | Параметры | Общие | Порты | Визуальные слои

Название	Имя	Формула	Значение
Коэффициенты усиления	k	2	[2]
Начальные условия	x0	6.46	[6.46]

Рис. 3.2.10

4. Настройка блока «Инерционное звено 1-го порядка» библиотеки «Динамические», рис. 3.2.11.

Свойства | Параметры | Общие | Порты | Визуальные слои

Название	Имя	Формула	Значение
Коэффициенты усиления	k	1	[1]
Постоянные времени	T	0.9	[0.9]
Начальные условия	x0	6.46	[6.46]

Рис. 3.2.11

5. Настройка блока «Идеальное транспортное запаздывание» библиотеки «Динамические», рис. 3.2.12.

Свойства | Общие | Порты | Визуальные слои

Название	Имя	Формула	Значение
Время запаздывания	tau	0.05	[0.05]

Рис. 3.2.12

6. Общие настройки параметров расчета, рис. 3.2.13.

Параметры расчёта | Синхронизация | Рестарт | База данных | Вид | Настройки

Название	Имя	Формула	Значение
Основные параметры			
Минимальный шаг	hmin		0.01
Максимальный шаг	hmax		0.01
Шаг синхронизации задачи в пакете	synstep		0.01
Режим параллельного выполнения в пакете	serial_mode		Параллельный
Начальный шаг интегрирования (если 0 - выбирается автоматически)	startstep		0
Метод интегрирования	intmet		Эйлера
Начальное время расчёта	starttime		0
Конечное время расчёта	endtime		8
Относительная ошибка	relerr		0.0001
Абсолютная ошибка	abserr		1E-6
Относительная ошибка сравнения времени для дискретных блоков и источников	time_rel_error		1E-12
Начальное значение неинициализированных выходов блоков	InitOutputsValue		0

Рис. 3.2.13

График переходного процесса представлен на рис. 3.2.14.

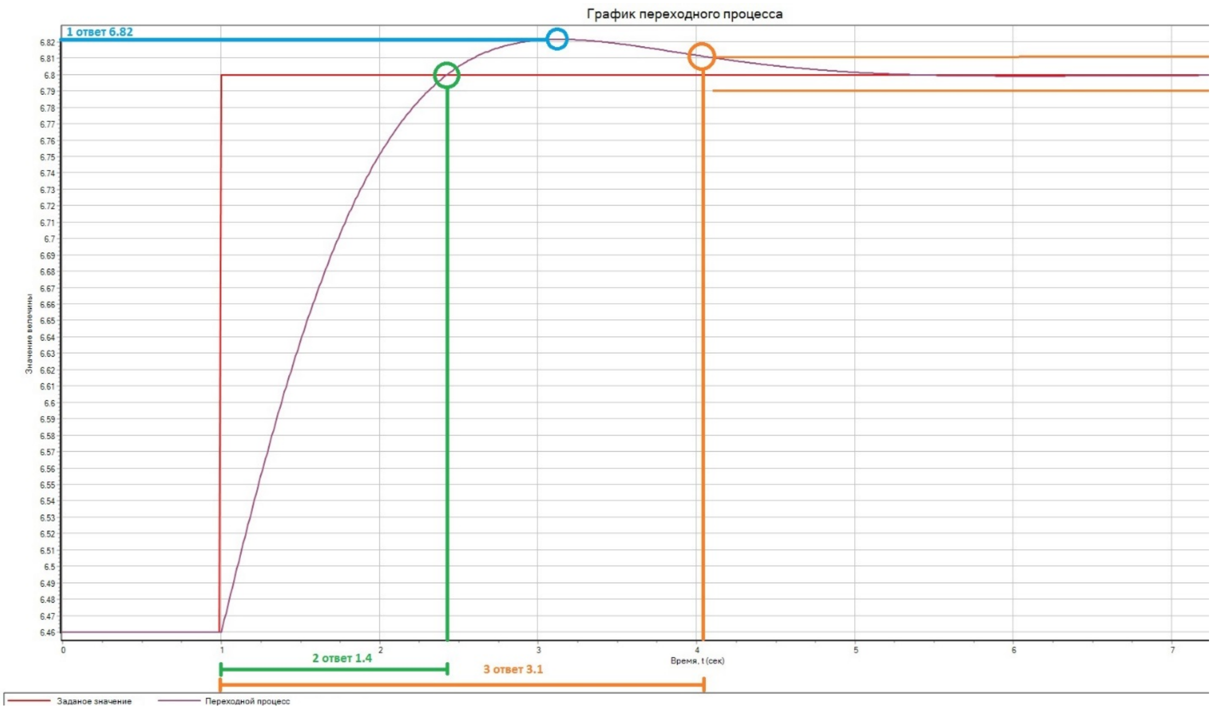

Рис. 3.2.14. График переходного процесса

Ответ:

Максимальное давление: 6,82 МПа.

Время первого достижения номинального значения: 1,4 с.

Время входа в трубку установившегося значения: 3,1 с.

4. Заключительный этап

4.1. Работа наставника НТО при подготовке к этапу

На этапе подготовки к заключительному этапу НТО наставник решает две важные задачи: помощь участникам в подготовке к предстоящим соревнованиям и формирование устойчивой и слаженной команды. Заключительный этап требует высокой слаженности, уверенности и глубоких знаний, и наставник становится тем, кто объединяет усилия участников и направляет их в нужное русло.

Наставник помогает участникам:

- разобрать задания прошлых лет, используя официальные сборники, чтобы понять структуру финальных испытаний, типы задач и ожидаемый уровень сложности;
- изучить организационные особенности заключительного этапа, включая формат проведения, регламент, продолжительность и технические нюансы;
- спланировать подготовку — на основе даты начала финала составляется четкий график занятий, в котором распределены темы, практикумы и командные тренировки;
- обратиться (при необходимости) за консультацией к разработчикам заданий по профилю, уточнить, на какие аспекты подготовки следует обратить особое внимание, и получить дополнительные материалы.

Также рекомендуется участие в мероприятиях от организаторов, таких как:

- установочные вебинары и открытые разборы задач;
- хакатоны, практикумы и мастер-классы для финалистов;
- встречи в онлайн-формате, информация о которых публикуется в группе НТО во «ВКонтакте» и в телеграм-чатах профилей.

Наставнику необходимо уделить внимание работе на формированием устойчивой, продуктивной и мотивированной команды:

- **Сплочение команды.** Это особенно актуально, если участники живут в разных городах. Регулярные онлайн-встречи, совместная работа над задачами и неформальное общение помогают наладить доверие и улучшить командную динамику.
- **Анализ ролей.** Наставник вместе с командой определяет, кто за что отвечает, какие задачи входят в зону ответственности каждого участника. Также обсуждаются возможности взаимозаменяемости на случай непредвиденных ситуаций.
- **Оценка компетенций.** Важно определить, какими знаниями и навыками уже обладают участники, а какие необходимо развить. На основе этого формируется индивидуальный и командный план подготовки.
- **Участие в подготовительных мероприятиях от разработчиков профилей.**

Перед заключительным этапом проводятся установочные вебинары, разборы задач прошлых лет, практикумы, мастер-классы для финалистов. Информация о таких мероприятиях публикуется в группе НТО в VK и в чатах профилей в Telegram.

- **Практика в формате хакатонов.** Наставник может организовать дистанционные хакатоны или практикумы с использованием заданий прошлых лет и методических рекомендаций из официальных сборников.

Таким образом, наставник становится координатором и моральной опорой команды, помогая пройти заключительный этап НТО с максимальной уверенностью и результатом.

4.2. Предметный тур

Задачи третьего этапа предметного тура профиля по информатике открыты для решения. Участие в соревновании доступно на платформе Яндекс.Контест: https://contest.yandex.ru/contest/72656/enter/.

4.2.1. Информатика. 8–11 классы

Задача 4.2.1.1. Экономный палиндром (10 баллов)

Имя входного файла: стандартный ввод или `input.txt`.

Имя выходного файла: стандартный вывод или `output.txt`.

Ограничение по времени выполнения программы: 1 с.

Ограничение по памяти: 64 Мбайт.

Условие

У Вениамина есть три **различные** маленькие буквы латинского алфавита. Вениамин хочет, чтобы из них можно было составить палиндром. Напомним, что строка называется палиндромом, если она читается одинаково как слева направо, так и справа налево. Например, строки `zzz`, `bob` являются палиндромами, а `abc` — не является. Очевидно, что из трех различных букв, как их ни переставляй, палиндром не получится.

Вениамин может удалить из своего набора одну любую букву и добавить в свой набор также любую букву. При этом как за удаление, так и за добавление он должен будет доплатить.

Для того чтобы узнать цену удаления или добавления буквы, пронумеруем их от 0 до 25: (буква a имеет номер 0, буква b имеет номер 1, ..., буква z имеет номер 25). Пусть изначально у Вениамина есть три буквы, которые обозначим α, β, γ, а сумму их номеров как S. Зафиксируем число S, оно в дальнейшем изменяться не будет ни при каких условиях.

Для любой буквы λ определим ее показатель $P(\lambda)$ как остаток от деления суммы номера этой буквы λ и числа S на 26.

Например, пусть у Вениамина есть три буквы d, e, z, (порядковые номера 3, 4 и 25). Тогда $S = 3 + 4 + 25 = 32$, а показатели, соответственно, следующие:

$$P(d) = (3 + 32)\%26 = 9,\ P(e) = (4 + 32)\%26 = 10,\ P(z) = (25 + 32)\%26 = 5.$$

Тогда для того чтобы добавить букву λ, нужно заплатить $P(\lambda)$, а для того чтобы удалить эту букву, нужно заплатить $26 - P(\lambda)$. Еще раз заметим, что показатель буквы вычисляется перед началом всех операций, и затем уже не изменяется.

Требуется вывести палиндром, который сможет в этих условиях построить Вениамин, заплатив за это минимальную возможную сумму.

Формат входных данных

В трех строках подаются три различных символа. Все они являются строчными буквами из интервала от a до z.

Формат выходных данных

Вывести в одну строку без пробелов палиндром из трех символов, который может быть получен из исходных букв путем удаления одной из них и добавления новой с наименьшими затратами. Можно показать, что при этих условиях решение единственно.

Примеры

Пример №1

Стандартный ввод
d e z

Стандартный вывод
zdz

Пример №2

Стандартный ввод
b a y

Стандартный вывод
byb

Пример программы-решения

Ниже представлено решение на языке C++.

C++

```cpp
#include<bits/stdc++.h>
#define sz(a) (int)a.size()
#define pb push_back
#define all(a) a.begin(), a.end()
#define for0(i, n) for(int i = 0; i < n; i++)
```

```
#define for1(i, n) for(int i = 1; i <= n; i++)
#define x first
#define y second
#define int long long
using namespace std;
typedef pair<int, int> pii;
typedef vector<int> vi;
const int INF = 1e18;
const int MOD = 1e9 + 7;
const int LG = 19;

int P(char a, int S){
    int res = (a - 'a' + S) % 26;
    return res;
}

int cost(char a, char b, int S){
    return P(a, S) + 26 - P(b, S);
}

signed main(){
        ios::sync_with_stdio(0), cin.tie(0), cout.tie(0);
    char a, b, c;
    cin >> a >> b >> c;
    int S = a + b + c - 3 * 'a';
    char ta = a, tb = b;
    int tcost = cost(a, b, S);
    if(cost(b, a, S) < tcost){
        ta = b, tb = a;
        tcost = cost(b, a, S);
    }
    if(cost(a, c, S) < tcost){
        ta = a, tb = c;
        tcost = cost(a, c, S);
    }
    if(cost(c, a, S) < tcost){
        ta = c, tb = a;
        tcost = cost(c, a, S);
    }
    if(cost(b, c, S) < tcost){
        ta = b, tb = c;
        tcost = cost(b, c, S);
    }
    if(cost(c, b, S) < tcost){
        ta = c, tb = b;
        tcost = cost(c, b, S);
    }
    char o = a + b + c - ta - tb;
    cout << ta << o << ta;
}
```

Задача 4.2.1.2. Былинная история (15 баллов)

Имя входного файла: стандартный ввод или `input.txt`.

Имя выходного файла: стандартный вывод или `output.txt`.

Ограничение по времени выполнения программы: 1 с.

Ограничение по памяти: 128 Мбайт.

Условие

Не все знают, но кроме короля Артура, круглый стол был и у великого былинного князя Владимира Красно Солнышко. За этим столом он любил собираться и пировать со своей многочисленной дружиной. Каждое место за этим столом имело свой номер, номера начинались с 1 и шли по часовой стрелке. Всего было n мест, и, так как стол был круглым, за последним n-м местом снова шло первое. Все члены дружины, включая князя, были упорядочены по старшинству, князь имел самый первый номер.

Рассадка производилась следующим образом: сначала рассаживались самые почетные члены дружины — первые k человек. Они выбирали случайные места по своему настроению и садились, как хотели. Далее возможность выбора места предоставлялась остальным $(n-k)$ членам дружины. Все они выбирали места по очереди в порядке старшинства. Каждый из них при выборе места действовал так: среди всех пустых оставшихся на данный момент мест выбиралось то, рядом с которым сидел самый почетный дружинник (то есть человек с самым маленьким номером), и садился на это место. Если у самого почетного, на текущий момент, оба места вокруг него были свободны, предпочтение отдавалось правому месту (выбиравший предпочитал сесть справа от выбранного почетного дружинника, это называлось «одесную»). С точки зрения нумерации мест это значило сесть на меньшее по номеру место (кроме случая, когда выбранный почетный дружинник сидел на первом месте).

Былины сохранили порядок в дружине князя и рассадку первых k ее членов. Нужно восстановить полную рассадку дружины, это важная историческая деталь.

Формат входных данных

В первой строке находится два целых числа n и k через пробел — общее количество человек в дружине, включая князя и количество самых почетных дружинников, которые могли выбирать места как хотят, $1 \leqslant k < n \leqslant 3 \cdot 10^5$.

Во второй строке через пробел указаны k чисел a_i — номера мест за столом, на которые сели эти k самых важных дружинников. Число a_i в этом наборе означает, что i-й дружинник сел на место с номером a_i, $1 \leqslant a_i \leqslant n$. Все a_i на входе попарно различны.

Формат выходных данных

Вывести $(n-k)$ чисел через пробел — номера мест, на которые сядут оставшиеся члены дружины. Номера выводятся в том порядке, в котором они выбирались, то есть первым выводится место, на которое сел $(k+1)$-й дружинник, вторым — место, на которое сел $(k+2)$-й и т. д.

Примеры

Пример №1

Стандартный ввод
10 3 5 4 1

Стандартный вывод
6 3 10 2 7 9 8

Примечания

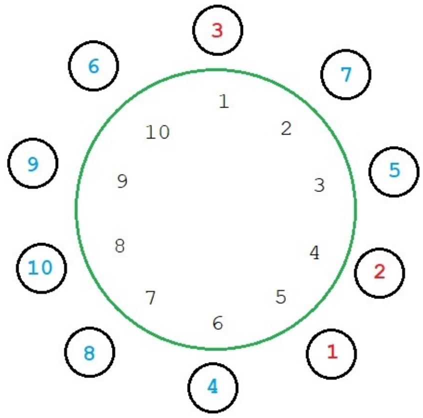

Рис. 4.2.1

На рисунке 4.2.1 представлена рассадка для примера из условия. Красными номерами указаны первые трое дружинников, которые выбирали места самостоятельно, синими номерами — остальные семеро, которые рассаживались по указанному в условии методу.

Пример программы-решения

Ниже представлено решение на языке C++.

C++

```cpp
#include<bits/stdc++.h>
#define sz(a) (int)a.size()
#define pb push_back
#define all(a) a.begin(), a.end()
#define for0(i, n) for(int i = 0; i < n; i++)
#define for1(i, n) for(int i = 1; i <= n; i++)
#define x first
#define y second
#define int long long
using namespace std;
typedef pair<int, int> pii;
typedef vector<int> vi;
```

```
const int INF = 1e18;
const int MOD = 1e9 + 7;
const int LG = 19;
signed main(){
        ios::sync_with_stdio(0), cin.tie(0), cout.tie(0);
    int n, k;
    cin >> n >> k;
    vi to(n, -1), pos(n, -1);
    for0(i, k){
        int a;
        cin >> a;
        a--;
        to[a] = i;
        pos[i] = a;
    }
    int t = 0;
    for(int i = k; i < n; i++){
        while(1){
            if(to[(pos[t] + n - 1) % n] == -1){
                cout << (pos[t] + n - 1) % n + 1 <<' ';
                to[(pos[t] + n - 1) % n] = i;
                pos[i] = (pos[t] + n - 1) % n;
                break;
            }
            if(to[(pos[t] + 1) % n] == -1){
                cout << (pos[t] + 1) % n + 1 <<' ';
                to[(pos[t] + 1) % n] = i;
                pos[i] = (pos[t] + 1) % n;
                break;
            }
            t++;
        }
    }
    cout << endl;
}
```

Задача 4.2.1.3. Плюс-минус

Имя входного файла: стандартный ввод или `input.txt`.

Имя выходного файла: стандартный вывод или `output.txt`.

Ограничение по времени выполнения программы: 1 с.

Ограничение по памяти: 64 Мбайт.

Условие

Возьмем некоторое натуральное число n и заведем $(n+1)$ ячейку. Пронумеруем ячейки от 0 до n. Изначально в каждой ячейке находится число 0. Мы находимся в ячейке с индексом 0 и начинаем двигаться в сторону увеличения индексов.

На каждом шаге перемещаемся в следующую по номеру ячейку и смотрим, какое число там стоит. Если это 0 или отрицательное число, тогда в ячейку с этим номером и во все кратные этому номеру ячейки прибавляется индекс этих ячеек. Если в ячейке с этим номером число больше 0, то из этой и всех кратных этому номеру ячеек вычитается их индекс. После этого пересчета переходим в следующую по номеру ячейку и повторяем набор описанных операций.

По числу n найти сумму всех чисел, оказавшихся в ячейках с номерами от 0 до n.

Например, $n = 8$.

Сначала находимся в 0, и все ячейки равны 0: 0, 0, 0, 0, 0, 0, 0, 0, 0.

Делаем шаг на ячейку номер 1. Там стоит 0. Тогда в нее и все кратные 1 (то есть вообще во все ячейки) добавим их индекс. Получим: 0, 1, 2, 3, 4, 5, 6, 7, 8.

Делаем шаг в ячейку номер 2. Там стоит 2, то есть число, большее 0. Вычитаем из нее и всех кратных 2 ячеек их индекс. Получим 0, 1, 0, 3, 0, 5, 0, 7, 0.

Делаем шаг в ячейку номер 3. Там стоит 3, то есть число, большее 0. Вычитаем из нее и всех кратных 3 ячеек их индекс. Получим 0, 1, 0, 0, 0, 5, -6, 7, 0.

Делаем шаг на ячейку номер 4. Там стоит 0. Тогда в нее и все кратные 4 добавим их индекс. Получим 0, 1, 0, 0, 4, 5, -6, 7, 8.

Делаем шаг в ячейку номер 5. Там стоит 5, то есть число, большее 0. Вычитаем из нее и всех кратных 5 ячеек их индекс. Получим 0, 1, 0, 0, 4, 0, -6, 7, 8.

Делаем шаг в ячейку номер 6. Там стоит -6, то есть число, меньшее 0. Прибавим к ней и во все кратные 6 ячейки их индекс. Получим 0, 1, 0, 0, 0, 5, 0, 7, 8.

Делаем шаг в ячейку номер 7. Там стоит 7, то есть число, большее 0. Вычитаем из нее и всех кратных 7 ячеек их индекс. Получим 0, 1, 0, 0, 4, 0, 0, 0, 8.

Делаем шаг в ячейку номер 8. Там стоит 8, то есть число, большее 0. Вычитаем из нее и всех кратных 8 ячеек их индекс. Получим 0, 1, 0, 0, 4, 0, 0, 0, 0.

Просуммируем получившиеся значения во всех ячейках и получим число 5. Это и будет ответом.

Формат входных данных

На вход подается единственное число n, $1 \leqslant n \leqslant 10^{13}$.

Формат выходных данных

Вывести одно число — итоговую сумму после того, как пройдем по всем n ячейкам.

Примеры

Пример №1

Стандартный ввод
8

Стандартный вывод
5

Пример программы-решения

Ниже представлено решение на языке C++.

C++

```cpp
#include<bits/stdc++.h>
#define int long long
using namespace std;
signed main(){
    ios::sync_with_stdio(0), cin.tie(0), cout.tie(0);
    int n;
    cin >> n;
    int L = sqrt(n);
    long double ans = 0;
    for(long double i = 1; i <= L; i++){
        ans += i * i;
    }
    cout.precision(0);
    cout << fixed<< ans;
}
```

Задача 4.2.1.4. Варианты эксперимента на БЛК (25 баллов)

Имя входного файла: стандартный ввод или `input.txt`.

Имя выходного файла: стандартный вывод или `output.txt`.

Ограничение по времени выполнения программы: 1 с.

Ограничение по памяти: 256 Мбайт.

Условие

Исследователи, работающие на БЛК (Большом линейном коллайдере), задумали новый эксперимент. На равных расстояниях в БЛК будет сгенерирован утвержденный набор частиц двух разных знаков. После старта эксперимента частицы полетят с одинаковой скоростью каждая в своем направлении (отрицательные влево, положительные — вправо). Если две частицы, летящие навстречу друг другу, сталкиваются, они взаимно уничтожаются. Эксперимент будет успешным, если в итоге взаимно уничтожатся все сгенерированные частицы.

Один из исследователей пишет диссертацию, в которой рассматривает все успешные варианты эксперимента. Вариантом эксперимента он называет некоторый непустой набор частиц, полученный из исходного путем удаления нескольких (возможно, ни одной) частиц из начала утвержденного набора и нескольких (возможно, ни одной) частиц из его конца. Вариант будет успешным, если эти частицы после старта снова все взаимно уничтожатся.

Для понимания объема своей диссертации ее автор хочет знать, сколько успешных вариантов эксперимента ему придется рассмотреть.

Формат входных данных

В единственной строке ввода содержится описание утвержденного набора частиц. Набор непустой. Каждая частица задается одной буквой `L` или `R`. Те, что помечены `L` после старта полетят влево, те что помечены `R` — вправо. Буквы в наборе заданы без пробелов между ними. Считаем, что соседние частицы расположены на равных расстояниях друг от друга. Гарантируется, что заданный на входе набор описывает успешный эксперимент (все частицы взаимно уничтожатся). Длина набора не превосходит 10^6.

Формат выходных данных

Вывести одно число — количество всех непустых успешных вариантов эксперимента для утвержденного набора частиц, заданного во входных данных.

Примеры

Пример №1

Стандартный ввод
RRLRRLRRLLLLRLRRLRLRRLLL

Стандартный вывод
20

Примечание.

Для примера из условия будет ровно 20 следующих успешных вариантов (числа указывают номер первой и последней частицы варианта): 2–3, 5–6, 8–9, 7–10, 5–10, 4–11, 2–11, 1–12, 13–14, 1–14, 16–17, 18–19, 16–19, 21–22, 20–23, 18–23, 16–23, 15–24, 13–24, 1–24.

Пример программы-решения

Ниже представлено решение на языке C++.

C++

```cpp
#include<bits/stdc++.h>
#define sz(a) (int)a.size()
#define pb push_back
#define all(a) a.begin(), a.end()
#define for0(i, n) for(int i = 0; i < n; i++)
#define for1(i, n) for(int i = 1; i <= n; i++)
#define int long long
using namespace std;
typedef pair<int, int> pii;
typedef vector<int> vi;
const int INF = 1e18;
const int MOD = 1e9 + 7;
const int LG = 19;

```

```
signed main(){
        ios::sync_with_stdio(0), cin.tie(0), cout.tie(0);
    string s;
    cin >> s;
    int ans = 0;
    vi st, dp(sz(s), 0);
    for0(i, sz(s)){
        if(s[i] == 'R'){
            st.pb(i);
        }
        if(s[i] == 'L'){
            dp[i]++;
            if(st.back() > 0 && s[st.back() - 1] == 'L'){
                dp[i] += dp[st.back() - 1];
            }
            st.pop_back();
        }
    }
    for0(i, sz(s)){
        ans += dp[i];
    }
    cout << ans << endl;
}
```

Задача 4.2.1.5. Пробка (30 баллов)

Имя входного файла: стандартный ввод или `input.txt`.

Имя выходного файла: стандартный вывод или `output.txt`.

Ограничение по времени выполнения программы: 1 с.

Ограничение по памяти: 128 Мбайт.

Условие

Предлагаем вам сыграть в увлекательную игру-головоломку «Пробка». Правила следующие: дано квадратное поле, разбитое на клетки, размер поля 6×6. Строки поля пронумеруем от 1 до 6 сверху вниз, столбцы от 1 до 6 слева направо. Поле ограничено по своему периметру ограждением. В этом ограждении, на его правой границе, напротив третьей строки есть свободный выход (выезд). Других выходов (выездов) в периметре нет.

На поверхности поля находятся несколько автомобилей. Каждый автомобиль занимает либо две, либо три рядом стоящие клетки в одном горизонтальном или вертикальном ряду. Таким образом, для автомобиля можно сказать, что он расположен горизонтально или вертикально. Автомобиль может передвигаться на свободную клетку, расположенную рядом с ним в его ряду (спереди или сзади от него). Автомобиль не может передвигаться вбок. В зависимости от вида автомобиля, будем называть его горизонтальным или вертикальным. Разумеется, никакие два автомобиля не могут одновременно находиться в одной клетке.

На поле имеется как минимум один горизонтальный автомобиль, назовем его автомобилем `A` или целевым автомобилем. Этот автомобиль всегда расположен в третьей строке. Кроме него в третьей строке нет ни одного горизонтального автомобиля.

Остальные автомобили могут располагаться произвольно, в том числе и по несколько в одном ряду.

Задача головоломки заключается в том, чтобы вывести из пробки автомобиль `A`, поставив его правую клетку на поле с координатами $(3, 6)$. Этому могут мешать другие автомобили. Их можно передвигать в пределах поля на свободные клетки в соответствии с их ориентацией. Одно передвижение одного любого автомобиля будем называть одним ходом.

Требуется за минимальное число ходов вывести `A`-автомобиль из пробки.

Формат входных данных

На вход в шести строках по шесть символов в каждой подается описание пробки. В этом описании пустому пространству соответствуют клетки, помеченные символом «.». Каждый автомобиль задается своей большой буквой латинского алфавита. Целевой автомобиль, который нужно вывести из пробки, всегда располагается в двух или трех рядом стоящих клетках третьего горизонтального ряда и обозначен буквой `A`. Остальные автомобили (если они есть) задаются своими большими буквами, отличными от буквы `A`. Буквы могут быть произвольными от `B` до `Z`. Каждый автомобиль занимает две или три рядом стоящие клетки в одном горизонтальном или вертикальном ряду.

Гарантируется, что всегда существует решение для заданной в тесте пробки.

Формат выходных данных

Требуется вывести одно число — минимальное количество ходов, требующихся для того, чтобы правая клетка автомобиля `A` оказалась в клетке с координатами $(3, 6)$.

Примеры

Пример №1

Стандартный ввод
``......```   ```......```   ```AAB..W```   ```..B..W```   ```Q.B..W```   ```QZZXXX```

Стандартный вывод
`10`

Примечание.

На рисунках 4.2.2–4.2.4 показан ход решения головоломки из примера. На первом рисунке показан начальный вид пробки. Далее по шагам показан кратчайший путь ее решения (кратчайших решений может быть несколько). Целевой автомобиль имеет красный цвет, в этом примере состоит из двух клеток и расположен в третьей строке. Видно, что за 10 ходов правая клетка целевого автомобиля попадает в клетку с координатой $(3, 6)$. Последний рисунок демонстрирует, что целевой автомобиль выехал из пробки. **Этот ход не нужно учитывать в ответе.**

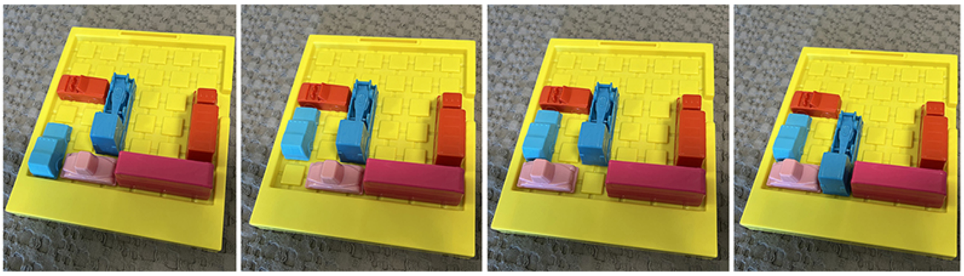

Рис. 4.2.2

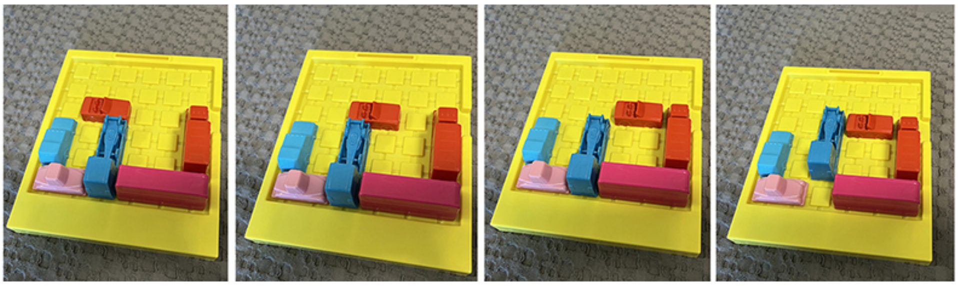

Рис. 4.2.3

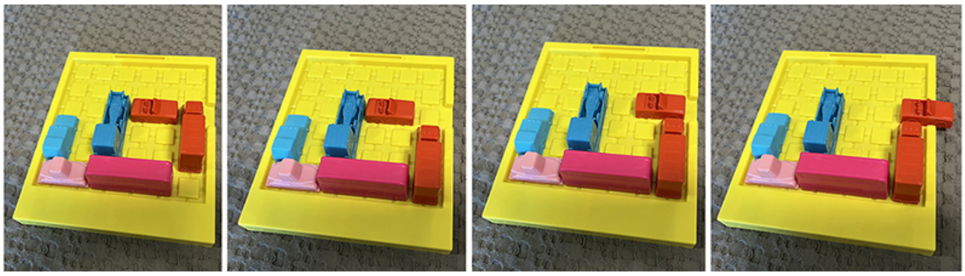

Рис. 4.2.4

Пример программы-решения

Ниже представлено решение на языке C++.

C++

```
#include<bits/stdc++.h>
#define sz(a) (int)a.size()
#define pb push_back
#define all(a) a.begin(), a.end()
#define for0(i, n) for(int i = 0; i < n; i++)
#define for1(i, n) for(int i = 1; i <= n; i++)
#define x first
#define y second
#define int long long
```

```
using namespace std;
typedef pair<int, int> pii;
typedef vector<int> vi;
typedef vector<vector<int> > vvi;
const int INF = 1e18;
const int MOD = 1e9 + 7;
const int LG = 19;

bool inside(int x, int y){
    return (x >= 0 && y >= 0 && x < 6 && y < 6);
}

void step_left(vector<string> &P, int x, int y){
    int t = y + 1;
    char c = P[x][y + 1];
    while(t < 6 && P[x][t] == c){
        t++;
    }
    t--;
    P[x][t] = '.';
    P[x][y] = c;
}

void step_right(vector<string> &P, int x, int y){
    int t = y - 1;
    char c = P[x][y - 1];
    while(t >= 0 && P[x][t] == c){
        t--;
    }
    t++;
    P[x][t] = '.';
    P[x][y] = c;
}

void step_up(vector<string> &P, int x, int y){
    int t = x + 1;
    char c = P[x + 1][y];
    while(t < 6 && P[t][y] == c){
        t++;
    }
    t--;
    P[t][y] = '.';
    P[x][y] = c;
}

void step_down(vector<string> &P, int x, int y){
    int t = x - 1;
    char c = P[x - 1][y];
    while(t >= 0 && P[t][y] == c){
        t--;
    }
    t++;
    P[t][y] = '.';
    P[x][y] = c;
}

signed main(){
    ios::sync_with_stdio(0), cin.tie(0), cout.tie(0);
    deque<vector<string> > D;
    map<vector<string>, int> used;
```

```
    map<vector<string>, vector<string> >par;
    vector<string> P(6);
    for0(i, 6){
        cin >> P[i];
    }
    par[P] = P;
    par[P][0][0] = '*';
    used[P] = 1;
    D.pb(P);
    while(sz(D) > 0)
    {
      vector<string> T = D[0], Z;
        if(T[2][5] == 'A'){
            vector<vector<string> > ans;
            vector<string> TT = T;
            while(TT[0][0] != '*'){
                ans.pb(TT);
                TT = par[TT];
            }
            reverse(all(ans));
            int cnt = 0;
            for(auto el : ans){
                cnt++;
            }
            cout << sz(ans) - 1 << endl;
            break;
        }
        D.pop_front();
        for0(i, 6){
            for0(j, 6){
                if(T[i][j] == '.'){
               if(inside(i, j + 1) && inside(i, j + 2) && T[i][j + 1] == T[i][j + 2]){
                        Z = T;
                        step_left(Z, i, j);
                        if(!used[Z]){
                            used[Z] = used[T] + 1;
                            D.pb(Z);
                            par[Z] = T;
                        }
                   }
                }

                if(T[i][j] == '.'){
               if(inside(i, j - 1) && inside(i, j - 2) && T[i][j - 1] == T[i][j - 2]){
                        Z = T;
                        step_right(Z, i, j);
                        if(!used[Z]){
                            used[Z] = used[T] + 1;
                            D.pb(Z);
                            par[Z] = T;
                        }
                   }
                }
                if(T[i][j] == '.'){
               if(inside(i + 1, j) && inside(i + 2, j) && T[i + 1][j] == T[i + 2][j]){
                        Z = T;
                        step_up(Z, i, j);
```

```
                        if(!used[Z]){
                            used[Z] = used[T] + 1;
                            D.pb(Z);
                            par[Z] = T;
                        }
                   }
                }
                if(T[i][j] == '.'){
               if(inside(i - 1, j) && inside(i - 2, j) && T[i - 1][j] == T[i - 2][j]){
                        Z = T;
                        step_down(Z, i, j);
                        if(!used[Z]){
                            used[Z] = used[T] + 1;
                            D.pb(Z);
                            par[Z] = T;
                        }
                   }
                }
            }
        }
    }
}
```

4.2.2. Физика. 8–9 классы

Задача 4.2.2.1. Слоеный пирог (16 баллов)

Для нужд микроэлектроники был изготовлен элемент, представляющий собой куб со стороной $a = 2$ мкм из $n = 32$ тонко напыленных слоев металла и $m = 31$ слоя слабопроводящего полупроводника, параллельных одной из пар граней куба. Удельное сопротивление полупроводника $r = 210$ Ом·м, удельное сопротивление металла ничтожно мало по сравнению с этой величиной. Плотность полупроводника $\rho_1 = 2\,000$ кг/м^3, плотность металла $\rho_2 = 8\,000$ кг/м^3. Определите сопротивление такого элемента в направлении поперек слоев, если его средняя плотность составила $\rho = 4\,000$ кг/м^3.

Решение

При измерении сопротивления слоистой структуры поперек слоев она включается в цепь как набор отдельных слоев, соединенных последовательно. Следовательно, ее общее сопротивление R может быть найдено по формуле

$$R = \sum_{i=1}^{n+m} R_i = \sum_{i=1}^{n+m} \frac{r_i d_i}{a^2},$$

где R_i — сопротивление i-го слоя, r_i — удельное сопротивление его материала, d_i — толщина этого слоя, а суммирование осуществляется по всем слоям структуры. Учитывая при этом, что сопротивление металлических слоев пренебрежимо мало по сравнению с сопротивлением полупроводниковых, можем суммировать только последние:

$$R = \sum_{i=1}^{n} R_i = \sum_{i=1}^{n} \frac{r d_i}{a^2} = \frac{r}{a^2} \sum_{i=1}^{n} d_i.$$

В этом случае удельное сопротивление r и площадь одного слоя a^2 выносятся за знак суммы, а оставшееся внутри него выражение представляет собой не что иное, как суммарную толщину всех полупроводниковых слоев, независимо от числа слоев. Обозначим ее b:

$$R = \frac{rb}{a^2}.$$

Чтобы найти эту суммарную толщину, выразим массу элемента через его среднюю плотность и через плотности отдельных его составляющих:

$$\rho a^3 = \rho_1 a^2 b + \rho_2 a^2 (a - b).$$

Сокращая a^2, получим линейное уравнение на b:

$$\rho a = \rho_1 b + \rho_2 (a - b) \Rightarrow b = \frac{\rho_2 - \rho}{\rho_2 - \rho_1} a,$$

откуда окончательно

$$R = \frac{r(\rho_2 - \rho)}{a(\rho_2 - \rho_1)} = 70\,\text{МОм}.$$

Ответ:

$$R = \frac{r(\rho_2 - \rho)}{a(\rho_2 - \rho_1)} = 70\,\text{МОм}.$$

Критерии оценивания

Указано, что слои можно считать соединенными последовательно или записано демонстрирующее это выражение для сопротивлений или напряжений и токов	3 балла
Показано, что сопротивление элемента определяется общей толщиной полупроводниковых слоев и не зависит от их числа и распределения	3 балла
Верно записано соотношение для средней плотности или общей массы, позволяющее найти общую толщину полупроводниковых слоев	4 балла
Получен правильный ответ	6 баллов
Всего	**16 баллов**

Задача 4.2.2.2. Оболочка (16 баллов)

При проведении ядерного эксперимента радиоактивный образец был окружен оболочкой поглотителей продуктов распада, представляющей собой сферический слой, имеющей внутренний радиус R и внешний радиус $4R$, в центре которого находился экспериментальный образец. Начальная температура всей оболочки $t_0 = 25$ °C.

После инициации в образце цепной реакции слой материала, находившийся от центра установки на расстоянии меньше $3R$, быстро поглотил большую энергию излучения и практически мгновенно достиг температуры $t_1 = 340$ °C, а материал, находившийся от центра установки на большем расстоянии, практически не изменил своей температуры. Определите, какая температура t_2 установилась в поглощающей оболочке через 20 мин, если известно, что за это время температура всей оболочки выровнялась, а внешней среде было передано $\eta = 20\%$ изначально поглощенной теплоты. Объем шара радиуса R задается выражением $4\pi R^3/3$.

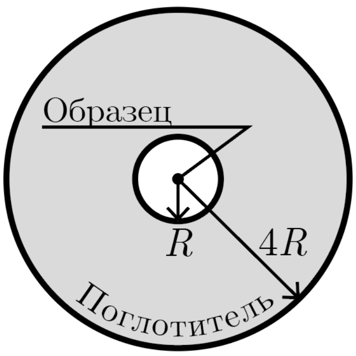

Рис. 4.2.5

Решение

Поглотитель представляет собой сферический слой, который можно представить как разность двух шаров, обладающих радиусами $4R$ и R.

Тогда общий объем V_0 поглотителя равен

$$V_0 = \frac{4\pi}{3}(4R)^3 - \frac{4\pi}{3}R^3 = 63\frac{4\pi}{3}R^3,$$

а объем V_1 нагревшейся части этого поглотителя

$$V_1 = \frac{4\pi}{3}(3R)^3 - \frac{4\pi}{3}R^3 = 26\frac{4\pi}{3}R^3.$$

Обозначив плотность поглотителя ρ, а его удельную теплоемкость c, легко найдем, что изначально веществом было поглощено количество теплоты

$$Q = c\rho V_1(t_1 - t_0) = 26\frac{4\pi}{3}c\rho R^3(t_1 - t_0).$$

По условиям задачи доля η этого тепла рассеивается во внешней среде, а остальная теплота перераспределяется равномерно по всему объему поглотителя:

$$Q(1-\eta) = c\rho V_0 = 63\frac{4\pi}{3}c\rho R^3(t_2 - t_0).$$

Подставляя эти два выражения друг в друга, получим

$$(1-\eta)26\cancel{\frac{4\pi}{3}c\rho R^3}(t_1 - t_0) = 63\cancel{\frac{4\pi}{3}c\rho R^3}(t_2 - t_0) \Rightarrow t_2 = t_0 + (1-\eta)\frac{26}{63}(t_1 - t_0) = 129\ ^\circ\text{C}.$$

Ответ:

$$t_2 = t_0 + (1-\eta)\frac{26}{63}(t_1 - t_0) = 129\ ^\circ\text{C}.$$

Критерии оценивания

Верно найден общий объем поглотителя (в том числе через неизвестный коэффициент пропорциональности)	3 балла
Верно найден объем нагревшейся части поглотителя (в том числе через неизвестный коэффициент пропорциональности)	2 балла
Записано правильное уравнение теплового баланса с учетом тепловых потерь	5 баллов
Получен правильный ответ	6 баллов
Всего	**16 баллов**

Задача 4.2.2.3. Деление (22 балла)

Покоящееся ядро нестабильного элемента, находящееся вдали от других заряженных частиц, претерпевает деление, в результате которого образуются два крупных осколка и два нейтрона. Первый осколок имеет массу $m_1 = 55$ а. е. м. и получает $\eta_1 = 0{,}8\%$ энергии, выделившейся в результате деления ядра, второй — массу $m_2 = 198$ а. е. м. и получает $\eta_2 = 0{,}5\%$ энергии, выделившейся в результате деления. Остальная энергия деления поровну распределилась между нейтронами. Вся энергия осколков чисто кинетическая. Под каким углом друг к другу разлетаются два нейтрона, если крупные осколки разлетелись под прямым углом друг к другу? Масса нейтрона $m_0 = 1$ а. е. м.

Решение

Импульс $\vec{p}$ частицы определяется выражением $\vec{p} = m\vec{v}$, где m — масса частицы, а $\vec{v}$ — ее скорость. Кинетическая энергия K частицы определяется выражением $K = mv^2/2$. Подставляя эти два выражения друг в друга, получим $p = \sqrt{2mK}$. Обозначим E общую энергию, выделившуюся в результате распада. Тогда

$$p_1 = \sqrt{2m_1\eta_1 E}; \quad p_2 = \sqrt{2m_2\eta_2 E}.$$

Суммарный импульс p_3 этих двух осколков легко найти, сложив $\vec{p_1} + \vec{p_2}$. Суммирование, разумеется, следует проводить геометрически, а значит, его модуль определяется по теореме Пифагора:

$$p_3 = \sqrt{p_1^2 + p_2^2} = \sqrt{2E(m_1\eta_1 + m_2\eta_2)}.$$

Импульс p_0 каждого из нейтронов легко выразить из той же логики, учитывая, что каждому из них достается половина остальной энергии:

$$p_0 = \sqrt{m_0 E(1 - \eta_1 - \eta_2)}.$$

Систему отсчета ядра можно считать инерциальной, поскольку по условиям ядро находится вдали от других заряженных частиц. Следовательно, геометрическая сумма импульсов всех четырех его осколков должна равняться нулю, а отдельно взятых двух нейтронов — составлять, соответственно, $-\vec{p_3}$. Из рис. 4.2.6 легко видеть, что модуль этой суммы равен

$$p_3 = 2p_0 \cos\frac{\phi}{2},$$

откуда получим

$$\cos\frac{\phi}{2} = \frac{p_3}{2p_0} = \frac{\sqrt{2\cancel{E}(m_1\eta_1 + m_2\eta_2)}}{2\sqrt{m_0\cancel{E}(1 - \eta_1 - \eta_2)}} \Rightarrow \phi = 2\arccos\sqrt{\frac{m_1\eta_1 + m_2\eta_2}{2m_0(1 - \eta_1 - \eta_2)}} \approx 63°.$$

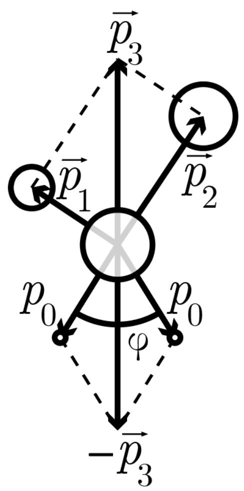

Рис. 4.2.6

Вообще говоря, шесть векторов на рис. 4.2.6 могут и не лежать в одной плоскости. Угол между импульсами нейтронов и тяжелых осколков установить невозможно, но внутри этих пар углы должны оставаться такими, какими они вычислены и изображены.

Ответ:

$$\phi = 2\arccos\sqrt{\frac{m_1\eta_1 + m_2\eta_2}{m_0(1-\eta_1-\eta_2)}} \approx 63°.$$

Критерии оценивания

Продемонстрировано понимание того, что импульс системы в рассматриваемом взаимодействии сохраняется	3 балла
Изображен рисунок или словами описана процедура векторного сложения импульсов	3 балла
Верно найдено выражение, связывающее импульс, массу и кинетическую энергию	4 балла
Грамотно применена теорема Пифагора к сложению импульсов тяжелых осколков	3 балла
Записано верное соотношение между импульсом одного нейтрона, суммарным импульсом двух и искомым углом	3 балла
Получен правильный ответ	6 баллов
Всего	**22 балла**

Задача 4.2.2.4. Поворот (26 баллов)

В ядерной лаборатории тестируются два различных устройства, предназначенных для управления пучком α-частиц. Оба устройства представляют собой кубические камеры одинаковых размеров со стороной l, в которые пучок α-частиц, испущенных радионуклидом, попадает через окошко в центре одной из граней параллельно оси симметрии камеры и должен выйти через противоположную стенку. Внутри камеры создано однородное поле, отклоняющее частицу, при этом интенсивность поля достаточна, чтобы ограничение на максимальный угол отклонения пучка частиц ϕ было обусловлено только геометрией камеры (см. рис. 4.2.7).

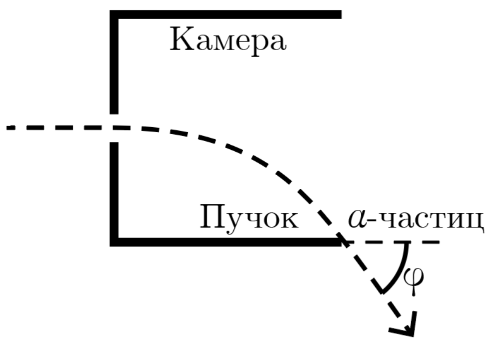

Рис. 4.2.7

Первое использует для поворота частицы электрическое поле, которое оказывает на частицу постоянную по величине и направлению силу, перпендикулярную вектору начальной скорости частицы. Второе — магнитное поле, оказывающее на частицу постоянную по величине силу, всегда направленную перпендикулярно вектору мгновенной скорости частицы На какую величину отличаются максимальные углы ϕ, на которые способны отклонить пучок в плоскости рис. 4.2.7 эти два устройства, если силы, создаваемые ими, существуют только в пределах кубической камеры?

Решение

Обозначим максимальные углы отклонения пучка ϕ_E и ϕ_M для электрической и магнитной камер соответственно. При движении в электрической камере под воздействием постоянной по модулю и направлению силы $\vec{F}$ закон движения частицы удобно записать в проекции на две оси: $0x$, сонаправленную с вектором ее начальной скорости и $0y$, сонаправленную с вектором $\vec{F}$ и создаваемым ей ускорением $\vec{a}$. Движение в первой проекции является равномерным, во второй — равноускоренным. При максимальном отклонении частица проходит у самого ребра кубической камеры, через точку с координатами $(l; l/2)$. Тогда для момента вылета частицы из камеры получим

$$\begin{cases} vt = l, \\ \dfrac{at^2}{2} = \dfrac{l}{2}. \end{cases}$$

Решая эту систему, получим $t = v/a$. Тогда вертикальная компонента скорости $v_y = at$, набранная частицей, равна $v_y = at = v$ ее изначальной горизонтальной компоненте скорости, а значит, угол ϕ_E максимального отклонения частицы составляет $45°$.

При движении в магнитной камере под действием силы, постоянно направленной перпендикулярно к вектору мгновенной скорости, созданное этой силой ускорение является чисто центростремительным, а значит, частица движется по дуге окружности. Найти радиус этой окружности можно при помощи геометрии.

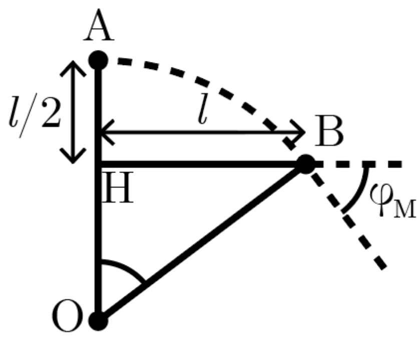

Рис. 4.2.8

Обозначим точку попадания частицы в камеру A, точку выхода частицы B, а центр кривизны ее траектории O. Также обозначим H угол в сечении камеры плоскостью рис. 4.2.8, лежащий на отрезке OA. Отрезки OA и OB равны как радиусы, обозначим их длины R. Тогда из геометрии камеры и теоремы Пифагора легко получить

$$(R - l/2)^2 + l^2 = R^2 \Rightarrow \frac{5}{4}l^2 - Rl = 0 \Rightarrow R = \frac{5l}{4},$$

откуда угол

$$\angle \text{AOB} = \arcsin \frac{4}{5},$$

но так как $\angle$OHB прямой, $\angle \text{AOB} = 90° - \angle \text{HBO} = \varphi_M$ (так как касательная к окружности перпендикулярна ее радиусу).

Тогда окончательно

$$\phi_M - \phi_E = \arcsin \frac{4}{5} - 45° \approx 8{,}1°.$$

Ответ:

$$\phi_M - \phi_E = \arcsin \frac{4}{5} - 45° \approx 8{,}1°.$$

Критерии оценивания

Продемонстрировано понимание того, что траектория α-частицы в электрической камере представляет собой дугу параболы	2 балла
Верно записаны законы движения частицы в этом случае	3 балла
Верно найдена вертикальная проекция скорости частицы на выходе из камеры	2 балла
Верно найден максимальный угол отклонения частицы в электрической камере	4 балла
Продемонстрировано понимание того, что траектория α-частицы в магнитной камере представляет собой дугу окружности	3 балла
Верно сделан чертеж, позволяющий найти угол отклонения в этом случае и/или записаны эквивалентные этому чертежу геометрические соотношения между отрезками	3 балла
Верно найден максимальный угол отклонения частицы в магнитной камере	5 баллов
Получен правильный ответ	4 балла
Всего	**26 баллов**

Задача 4.2.2.5. Фонарик (20 баллов)

Для медицинских нужд изготавливается микроскопический осветительный прибор, работающий внутри биологических тканей, окружающая среда в которых может иметь достаточно высокий показатель преломления. Его испытания проводятся в специальном растворе с показателем преломления $n_0 = 1{,}5$. Прибор представляет собой крошечный светодиод, напаянный на плоскую плату и окруженный «капелькой» из оптической смолы с низким показателем преломления $n < n_0$. Капелька имеет форму сегмента сферы, сходящегося с плоскостью платы под углом $\alpha = 15°$, как изображено на рис. 4.2.9. Определите значение n, если известно, что устроенный таким образом фонарик внутри раствора дает конус света, угол между крайними лучами которого $\phi = 120°$.

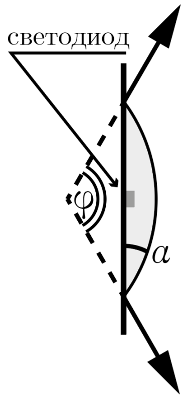

Рис. 4.2.9

Решение

Чем ближе к полюсу капельки падает луч, тем меньше будет угол его падения, а значит, по закону Снеллиуса, меньше будет и угол преломления. Поэтому угол ϕ раствора светового конуса будет определяться преломлением крайних лучей, изображенных на рис. 4.2.10. До преломления эти лучи распространяются практически параллельно плате, поэтому их угол падения β на границу капельки и крови равен $\beta = 90° - \alpha$. Изобразим ход этих лучей на чертеже вблизи. На чертеже О — центр кривизны капельки, А — точка преломления луча, В — вершина светового конуса, С — светодиод.

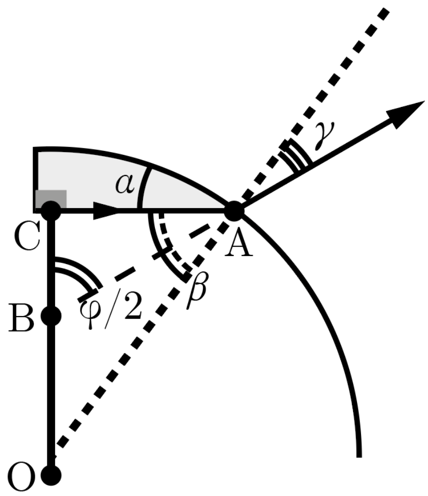

Рис. 4.2.10

Тогда, в соответствии с законом Снеллиуса, угол преломления γ этих лучей связан с углом падения выражением

$$n_0 \sin\gamma = n \sin\beta = n \cos\alpha \Rightarrow n = \frac{\sin\gamma}{\cos\alpha} n_0.$$

В треугольнике ABC угол при вершине A равен $\beta - \gamma$. Тогда из суммы углов этого треугольника легко получить

$$\beta - \gamma + 90° + \phi/2 = 180° \Rightarrow \gamma = \beta + \phi/2 - 90° = \phi/2 - \alpha.$$

Подставляя этот результат в закон Снеллиуса, получим

$$n = \frac{\sin(\phi/2 - \alpha)}{\cos\alpha} n_0 \approx 1{,}1.$$

Ответ:

$$n = \frac{\sin(\phi/2 - \alpha)}{\cos\alpha} n_0 \approx 1{,}1.$$

Изображен ход крайнего луча с отметкой углов его падения и преломления	3 балла
Верно записан закон Снеллиуса	4 балла
Верно найдено соотношение между углом α и углом падения луча	2 балла
Верно найдено соотношение между углами α, ϕ и углом преломления луча	5 баллов
Получен правильный ответ	6 баллов
Всего	**20 баллов**

4.2.3. Физика. 10–11 классы

Задача 4.2.3.1. Диполь (15 баллов)

Электрический диполь, представляющий собой пару равных по модулю разноименных зарядов, которые соединены жесткой связью на некотором расстоянии l друг от друга, влетает со скоростью $v = 200$ м/с в скрещенные под прямым углом друг к другу электрическое и магнитное поля. При этом скорость диполя сонаправлена с силовыми линиями электрического поля, а вектор $\vec{l}$, соединяющий два заряда диполя, перпендикулярен линиям индукции магнитного (см. рис. 4.2.11).

Под каким углом к силовым линиям электрического поля должен быть ориентирован диполь, чтобы его движение оставалось чисто поступательным? Напряженность электрического поля $E = 400$ В/м, индукция магнитного $B = 0{,}5$ Тл.

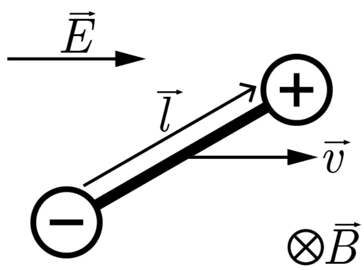

Рис. 4.2.11

Решение

В электрическом поле на положительный заряд действует сила Кулона $\vec{F}_E = \vec{E}q$, направленная вправо на рис. 4.2.11, а на отрицательный — строго противоположная ей. В магнитном поле на положительный заряд действует сила $F_M = qvB$, направленная (по правилу левой руки) вверх на рис. 4.2.11, а на отрицательный — строго противоположная ей.

Поскольку обе эти силы возникают равными по модулю и противоположными по направлению парами, ускорение диполя в любом случае равно нулю, и он представляет собой инерциальную систему отсчета. Чтобы его движение было чисто поступательным, нулю также должна равняться сумма моментов этих сил относительно любой оси.

Проще всего записать соответствующие условия относительно оси, проходящей перпендикулярно плоскости рис. 4.2.11 через отрицательный заряд. В этом случае момент силы F_E равен

$$M_E = F_E l \sin\alpha,$$

где α — искомый угол между векторами $\vec{l}$ и скорости диполя, а момент силы F_M аналогично равен

$$M_M = F_M l \cos\alpha.$$

Приравнивая эти два момента, получим

$$qEl\sin\alpha = qvBl\cos\alpha \Rightarrow \operatorname{tg}\alpha = \frac{vB}{E} \Rightarrow \alpha = \operatorname{arctg}\frac{vB}{E} \approx 14°.$$

Ответ:

$$\alpha = \operatorname{arctg}\frac{vB}{E} \approx 14°.$$

Критерии оценивания

Верно записано выражение для силы Кулона, действующей на заряды и определено направление этой силы	2 балла
Верно записано выражение для силы Лоренца, действующей на заряды и определено направление этой силы	4 балла
Верно записано уравнение моментов	5 баллов
Получен правильный ответ	4 балла
Всего	**15 баллов**

Задача 4.2.3.2. Плазменный канал (17 баллов)

В лаборатории исследуется возможность передачи электрического разряда на большие расстояния по плазменному каналу в атмосфере. Для этого мощный пучок параллельных рентгеновских лучей «прожигает» по ходу своего следования воздух, в среднем отрывая от каждой молекулы газа по одному электрону, после чего высоковольтный источник создает в полученном таким образом канале однородное электрическое поле с напряженностью. Найдите удельное сопротивление полученного таким образом плазменного канала, если известно, что скорость $\vec{v}$ дрейфа электронов в ионизированном воздухе прямо пропорциональна напряженности $\vec{E}$ электрического поля $v = -\mu\vec{E}$ с коэффициентом $\mu = 0{,}15$ Кл·с/кг (называемым подвижностью), а подвижность положительных ионов пренебрежимо мала. Концентрация молекул воздуха в условиях эксперимента $n = 2{,}6 \cdot 10^{25}$ м$^{-3}$.

Решение

Удельное сопротивление вещества ρ может быть найдено по формуле

$$\rho = \frac{RS}{l},$$

где R — электрическое сопротивление рассматриваемого объема вещества, S — площадь его поперечного сечения, l — длина. Дополняя эту формулу определением электрического сопротивления $R = U/I$, где U — напряжение на концах элемента, а I — сила тока в нем, получим

$$\rho = \frac{US}{Il}.$$

Сила тока I, равная отношению переносимого через поперечное сечение проводника заряда q ко времени t, может быть выражена через скорость движения заряженных частиц v, заряд e одной из них и их концентрацию n как

$$I = \frac{q}{t} = \frac{enV}{t} = \frac{enSl}{t} = enSv.$$

Здесь учитываем, что вклад ионов в общую проводимость плазменного канала пренебрежимо мал по сравнению со вкладом электронов, поскольку по условиям задачи пренебрежимо мала их подвижность и, следовательно, дрейфовая скорость. Тогда

$$\rho = \frac{U\cancel{S}}{en\cancel{S}vl} = \frac{E}{env} = \frac{1}{en\mu} \approx 1{,}6\,\text{мкОм·м},$$

где E — напряженность электрического поля, связанная в однородном поле с напряжением соотношением $U = El$.

Ответ:

$$\rho = \frac{1}{en\mu} \approx 1{,}6\,\text{мкОм·м}.$$

Критерии оценивания

Верно записано выражение связи удельного сопротивления с обычным	3 балла
Верно записан закон Ома	3 балла
Верно записана связь силы тока со скоростью, зарядом и концентрацией зарядов в проводнике	6 баллов
Получен правильный ответ	5 баллов
Всего	**17 баллов**

Первые два пункта критериев также могут быть заменены записью закона Ома в дифференциальной форме, если он известен участникам.

Задача 4.2.3.3. Наноантенна (20 баллов)

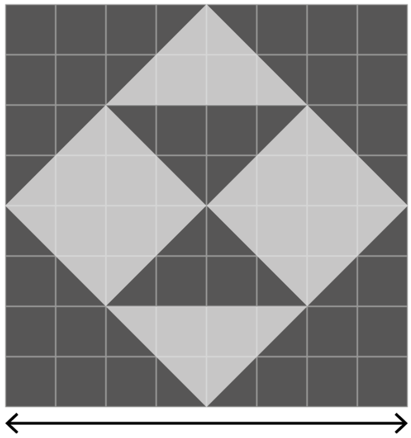

Рис. 4.2.12

Наноантенны представляют собой миниатюрные электронные устройства, структурированные в наномасштабе комбинации различных материалов. Зачастую они имеют причудливые формы, сконструированные для взаимодействия со строго определенными типами излучения.

В лаборатории была изготовлена антенна, форма которой изображена на рис. 4.2.12: более темные участки представляют собой диэлектрик с проницаемостью $\varepsilon_1 = 6$, более светлые — диэлектрик с проницаемостью $\varepsilon_2 = 2$. Клетчатый узор представляет собой исключительно масштабную линейку, каждая клетка которой имеет сторону $a = 100$ нм. Изображенная структура изготовлена в виде плоского элемента с толщиной $d = 20$ нм (в направлении, перпендикулярном плоскости рис. 4.2.12), зажатого между двумя такими же тонкими слоями металла. Оцените электроемкость полученного таким образом элемента. Электрическая постоянная $\varepsilon_0 = 8{,}85 \cdot 10^{-12}$ Ф/м.

Решение

Из условий легко видеть, что толщина d диэлектрического слоя между двумя металлическими много меньше всех продольных характерных размеров узора антенны, наименьший из которых составляет порядка $10d$.

В этом случае систему из двух тонких металлических слоев, между которыми зажат диэлектрический, можно рассматривать как плоский конденсатор. Причем каждую фигуру на изображенном рисунке антенны можно рассматривать как отдельный конденсатор, емкость которого задается формулой

$$C = \frac{S\varepsilon\varepsilon_0}{d},$$

где S — площадь этого участка, ε — диэлектрическая проницаемость образующего его диэлектрика.

Площадь каждого элемента узора легко найти «по клеточкам». На рис. 4.2.13 она подписана в единицах $a^2 = 10^{-14}\text{м}^2$ (площадь одной клетки). Все эти независимые конденсаторы соединены параллельно, поэтому общая емкость антенны может быть найдена как сумма их емкостей:

$$C = \frac{\varepsilon_0}{d} \sum_{i=1}^{10} S_i \varepsilon_i,$$

где i — номер участка на рис. 4.2.13 антенны, S_i — его площадь, а ε_i — его диэлектрическая проницаемость.

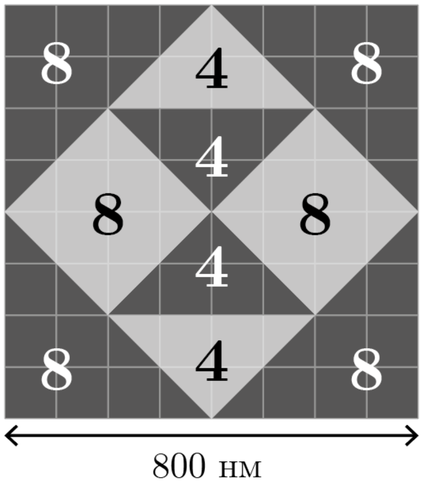

Рис. 4.2.13

Подставляя в это выражение площади, отмеченные на рис. 4.2.13, и проницаемости, данные в условиях, получим окончательно

$$C = \frac{a^2 \varepsilon_0}{d}(40\varepsilon_1 + 24\varepsilon_2) \approx 1{,}27 \cdot 10^{-15}\,\Phi.$$

Ответ:

$$C = \frac{a^2 \varepsilon_0}{d}(40\varepsilon_1 + 24\varepsilon_2) \approx 1{,}27 \cdot 10^{-15}\,\Phi.$$

Критерии оценивания

Верно записано выражение для электроемкости плоского конденсатора	4 балла
Обоснована его применимость в связи с соотношением геометрических размеров системы	3 балла
Замечено, что емкости отдельных фигур антенны выступают как соединенные параллельно	4 балла
Верно найдены площади всех необходимых участков антенны	3 балла
Получен правильный ответ	6 баллов
Всего	**20 баллов**

Задача 4.2.3.4. Благородные газы (26 баллов)

Радиоизотоп X делится с периодом полураспада $T_1 = 20$ сут., в результате чего образуется ядро аргона $^{40}_{18}\mathrm{Ar}$ и стабильное металлическое ядро. Радиоизотоп Y с периодом полураспада $T_2 = 8$ сут. претерпевает α-распад, в ходе которого образуется еще один α-радиоактивный изотоп Z, период полураспада которого составляет несколько микросекунд. В результате распада изотопа Z также образуется стабильное металлическое ядро. Одинаковые количества вещества изотопов X и Y помещают в камеру, из которой откачан газ. По прошествии $t = 40$ сут. оказывается, что в этой камере образовалась смесь благородных газов, при температуре $\theta = 25$ °C и давлении $p = 12$ кПа. Найдите плотность этой смеси. Универсальная газовая постоянная $R \approx 8{,}31$ Дж/(моль·K), абсолютный ноль температур $0\,\mathrm{K} = -273$ °C. α-частица представляет собой ядро $^4_2\mathrm{He}$.

Решение

В результате одного деления ядра изотопа X образуется один атом аргона, следовательно, количество вещества ν_{Ar} этого газа связано с числом N_1 произошедших делений соотношением

$$\nu_{Ar} = N_1/N_A,$$

где N_A — число Авогадро.

В результате одного α-распада ядра изотопа Y образуется один атом гелия, а также сильно нестабильный изотоп, который практически сразу (в течение микросекунд) претерпевает еще один α-распад, также с образованием одного атома гелия. Поэтому количество ν_{He} этого газа связано с числом N_2 произошедших первичных α-распадов соотношением

$$\nu_{He} = 2N_2/N_A.$$

Согласно закону радиоактивного распада, при распаде с периодом полураспада T число n нераспавшихся ядер по прошествии времени t задается выражением $n = n_0 2^{-t/T}$, где n_0 — изначальное число этих ядер. Следовательно, число N распадов, произошедших за это время, задается выражением

$$N = n_0 - n = n_0(1 - 2^{-t/T}).$$

Используя это выражение, легко выразим количества вещества обоих образовавшихся газов:

$$\nu_{Ar} = \frac{n_0}{N_A}(1 - 2^{-t/T_1}) = \frac{3n_0}{4N_A}; \quad \nu_{He} = 2\frac{n_0}{N_A}(1 - 2^{-t/T_2}) = \frac{31n_0}{16N_A}.$$

Для смеси газов, образовавшихся в камере, справедлив закон Дальтона

$$p = p_{Ar} + p_{He},$$

где p_{Ar}, p_{He} — парциальные давления аргона и гелия соответственно. Каждый из этих газов подчиняется уравнению Менделеева – Клапейрона, принимающему вид

$$pV = (\nu_{Ar} + \nu_{He})R\theta,$$

где V — объем камеры. Выразим его через молярные массы газов и искомую плотность смеси:

$$V = \frac{\mu_{Ar}\nu_{Ar} + \mu_{He}\nu_{He}}{\rho} \Rightarrow \frac{p}{\rho}(\mu_{Ar}\nu_{Ar} + \mu_{He}\nu_{He}) = (\nu_{Ar} + \nu_{He})R\theta.$$

Выражая из этого уравнения ρ, получим

$$\rho = \frac{p(\mu_{Ar}\nu_{Ar} + \mu_{He}\nu_{He})}{(\nu_{Ar} + \nu_{He})R\theta} = \frac{p\left(\mu_{Ar}\left(\frac{3n_0}{4N_A}\right) + \mu_{He}\left(\frac{31n_0}{16N_A}\right)\right)}{\left(\frac{3n_0}{4N_A} + \frac{31n_0}{16N_A}\right)R\theta}.$$

Сокращая дробь на n_0/N_A и приводя все отношения к общему знаменателю, получим окончательно

$$\rho = \frac{p(12\mu_{Ar} + 31\mu_{He})}{43R\theta} \approx 68\,\text{г/м}^3.$$

Молярные массы аргона и гелия учащиеся должны понять из символов $^{40}_{18}$Ar, $^{4}_{2}$He, данных в условиях.

Ответ:

$$\rho = \frac{p(12\mu_{Ar} + 31\mu_{He})}{43R\theta} \approx 68\,\text{г/м}^3.$$

Критерии оценивания

Верно записан хотя бы один закон радиоактивного распада	3 балла
Замечено и обосновано, что количество ядер гелия вдвое превосходит число первичных α-распадов	3 балла
Верно найдены количества образовавшихся газов или их отношение	5 баллов
Верно записан закон Дальтона	3 балла
Верно записано уравнение Менделеева – Клапейрона или основное уравнение МКТ в любой форме	3 балла
Уравнение Менделеева – Клапейрона или основное уравнение МКТ преобразовано к (или сразу записано в) форме, содержащей искомую плотность	3 балла
Получен правильный ответ	6 баллов
Всего	**26 баллов**

Задача 4.2.3.5. Радиационное трение (22 балла)

Согласно законам электродинамики, заряженные частицы, движущиеся с ускорением, испытывают взаимодействие со своими собственными электромагнитными полями, называемое радиационным трением. В общем случае описание этого взаимодействия весьма трудоемко, но для относительно медленных частиц сила радиационного трения может быть найдена по формуле $\vec{F} = \dfrac{q^2}{6\pi\varepsilon_0 c^3}\dfrac{\Delta\vec{a}}{\Delta t}$, где q — электрический заряд частицы, c — скорость света в вакууме, ε_0 — электрическая постоянная, $\Delta\vec{a}/\Delta t$ — мгновенная скорость изменения мгновенного ускорения $\vec{a}$-частицы. В эксперименте, действуя на частицу с зарядом $q < 0$ и массой m внешним электромагнитным полем, требуется поддерживать ее равномерное движение по окружности радиуса R. При какой угловой скорости частицы сила, действующая на нее со стороны этого внешнего поля для обеспечения такого движения, должна быть направлена под углом 45° к скорости частицы?

Решение

В приведенном выражении для силы радиационного трения затруднения может вызвать только дробь $\Delta\vec{a}/\Delta t$. Эта величина — «мгновенная скорость» изменения ускорения — находится с мгновенным ускорением в том же соотношении, в котором само ускорение находится с мгновенной скоростью, а мгновенная скорость — с радиус-вектором.

При равномерном движении по окружности радиус-вектор $\vec{r}$, проведенный в точку из центра траектории, имеет постоянный модуль R и вращается вокруг своего начала с некоторой постоянной угловой скоростью ω. Из школьной кинематики известно, что вектор мгновенной скорости $\vec{v}$ при таком движении имеет модуль $v = \omega R$ и также вращается с угловой скоростью ω, «опережая» радиус-вектор на 90° в направлении вращения, поскольку направлен по касательной к траектории. Аналогично вектор мгновенного (центростремительного) ускорения $\vec{a}$ имеет модуль $a = \omega v = \omega^2 R$ и вращается с угловой скоростью ω, «опережая» вектор мгновенной скорости на 90° в направлении вращения, поскольку направлен к центру траектории.

Следовательно, используя те же математические инструменты, что доказывают предыдущие две связи, можно показать, что вектор $\Delta\vec{a}/\Delta t$ имеет модуль $\omega a = \omega^3 R$ и вращается с угловой скоростью ω, «опережая» вектор центростремительного ускорения на 90° в направлении вращения. Поскольку вне зависимости от знака заряда выражение q^2 в числителе формулы для радиационного трения больше нуля, как больше нуля и все входящие в выражение физические константы, вектор $\vec{F}$ силы радиационного трения направлен так же, как и $\Delta\vec{a}/\Delta t$ — строго против вектора мгновенной скорости, по касательной к траектории (см. рис. 4.2.14).

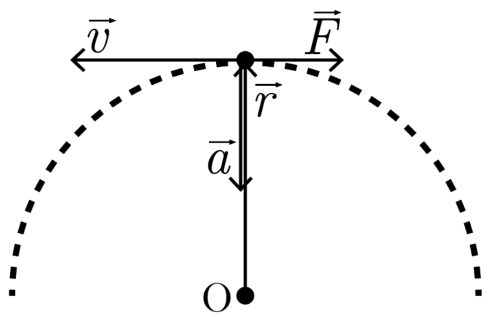

Рис. 4.2.14

Чтобы ответить на вопрос задачи, заметим, что для поддержания равномерного движения заряженной частицы по окружности внешняя сила $\vec{F}_{\text{вн}}$ должна

- создавать центростремительное ускорение,
- компенсировать радиационное трение.

Составляющая $F_{\text{вн}\|}$ этой силы, параллельная скорости, должна быть равна по модулю силе радиационного трения:

$$F_{\text{вн}\|} = \frac{q^2}{6\pi\varepsilon_0 c^3}\omega^3 R,$$

в то время как составляющая $F_{\text{вн}\perp}$ этой силы, перпендикулярная скорости, должна создавать центростремительное ускорение и, следовательно, равна

$$F_{\text{вн}\perp} = m\omega^2 R.$$

Поскольку по условиям сила $\vec{F}_{\text{вн}}$ направлена к скорости под углом 45°, составляющие $F_{\text{вн}\|}$ и $F_{\text{вн}\perp}$ этой силы равны:

$$\frac{q^2}{6\pi\varepsilon_0 c^3}\omega^3 r = m\omega^2 r \Rightarrow \omega = \frac{6\pi\varepsilon_0 mc^3}{q^2}.$$

Ответ:

$$\omega = \frac{6\pi\varepsilon_0 mc^3}{q^2}.$$

Продемонстрировано понимание того, что вектор $\Delta\vec{a}/\Delta t$ относится к центростремительному ускорению так же, как оно само относится к мгновенной скорости, а та — к радиус-вектору	4 балла
Верно найдено выражение для модуля этого вектора через любой набор более стандартных величин кинематики: линейную и/или угловую скорость или центростремительное ускорение, радиус	3 балла
Верно найдено направление этого вектора	3 балла
Продемонстрировано понимание того, что знак заряда в условиях не играет роли для ответа	2 балла
Продемонстрировано понимание того, что направление внешней силы под углом 45° к скорости означает равенство сил, необходимых для преодоления радиационного трения и для создания центростремительного ускорения	4 балла
Верно найден ответ на вопрос задачи	6 баллов
Всего	**22 балла**

Задача, разумеется, также имеет решение через производные. В этом случае запись верного определения вектора $\dot{\vec{a}}$ как производную ускорения по времени является частным случаем выполнения первого критерия, а правильное вычисление этой производной по координатам — второго.

4.3. Инженерный тур

4.3.1. Общая информация

В финальной задаче участникам предлагается на рассмотрение реальная технологическая проблема по обеспечению безопасности и надежности функционирования паротурбинной установки на атомных станциях. В процессе решения нужно разработать и реализовать в программной платформе SimInTech цифровую модель системы управления турбоустановкой для реактора ВВЭР с визуализацией протекающих процессов с помощью интерактивного человеко-машинного интерфейса, а также провести ряд модельных экспериментов с целью подготовки алгоритмов управления и технологических защит турбины.

4.3.2. Легенда задачи

Во время наблюдения за параметрами оборудования турбинного отделения АЭС ведущий инженер управления турбиной (ВИУТ) заметил, что один из датчиков мощности системы управления тихоходной паровой турбины периодически показывает некорректные значения.

После изучения архива сотрудник цеха тепловой автоматики и измерений (ЦТАИ) обнаружил, что кратковременные сбои в работе датчика приводят к нежелательной реакции системы управления в виде управляющего воздействия на регулирующие клапаны турбины с последующим изменением мощности турбогенератора.

Для тщательного изучения влияния отказов датчиков на систему управления и исключения любой возможности повреждения оборудования турбины было организовано совместное совещание с командой разработчиков системы.

В результате совещания сотрудников ЦТАИ и разработчиков системы управления было принято решение о проведения испытаний системы регулирования и защит турбины при отказах датчиков мощности в режиме работы АЭС на номинальной мощности после завершения начального нагружения. Для этого следует создать упрощенную математическую модель турбины с регулятором мощности и элементами системы верхнего уровня.

4.3.3. Требования к команде и компетенциям участников

Количество участников в команде: 3–4 человека.

Компетенции, которыми должны обладать члены команды:

1. **Математик**: математическое обеспечение разработки.
2. **Программист**: программная реализация цифровой модели.

3. **Инженер-физик**: знание основ физики ядерных реакторов, умение ориентироваться в составе и оборудовании АЭС.

4.3.4. Оборудование и программное обеспечение

Таблица 4.3.1

Наименование	**Описание**
SimInTech	Для разработки математических моделей, алгоритмов управления, мнемосхем и видеокадров

4.3.5. Описание задачи

Создание упрощенной математической модели турбины АЭС для реакторной установки ВВЭР с пультом управления СВБУ.

В командной задаче предлагается:

1. Подготовить математическую модель системы управления паровой тихоходной турбины АЭС с реактором ВВЭР.
2. Реализовать цифровую модель в специализированном программном обеспечении SimInTech.
3. Разработать графический информационно-управляющий интерфейс.
4. Реализовать систему автоматического регулирования мощности паровой турбины с элементами системы защит от недопустимого разгона.
5. Провести исследования объекта управления и испытания системы регулирования и защит паровой турбины при помощи полученного цифрового тренажера.

Описание объекта управления

Основные расчетные характеристики турбоустановки:

Номинальная электрическая мощность — 1 200 МВт.

Рабочая частота вращения — 1 500 об/мин.

Упрощенная математическая модель турбины состоит из трех основных элементов:

- паровые отсеки проточной части турбины;
- ротор турбоустановки;
- генератор.

Проточная часть паровой турбины АЭС содержит два цилиндра (цилиндр высокого давления ЦВД и цилиндр низкого давления ЦНД) с промежуточным перегревом пара между ними. Ниже (см. рис. 4.3.1) представлена общая схема проточной части паровой турбины, состоящая из ЦВД и ЦНД с трактом промежуточного перегрева пара в сепараторе пароперегревателе (СПП).

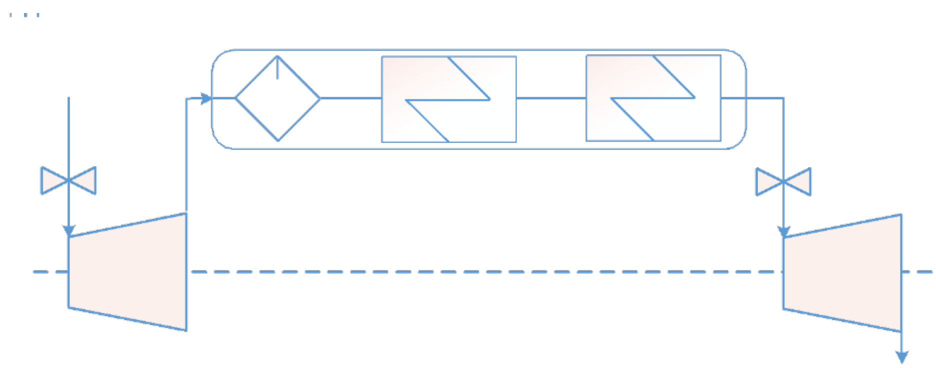

Рис. 4.3.1. Общая схема проточной части турбины

Динамика процесса изменения мощности и момента на валу паровой турбины при изменении расхода пара определяется, главным образом, паровыми объемами между регулирующими клапанами и соответствующими соплами ступеней турбины, а также паровыми объемами в тракте промежуточного перегрева пара.

При изменении расхода пара через турбину ЦВД быстро корректирует свою мощность, а ЦНД — медленно. Запаздывание создает большой буферный объем пара в тракте промежуточного перегрева, значительно ухудшающий динамические характеристики турбины.

Такое поведение динамических процессов в проточной части паровой турбины можно воспроизвести с помощью следующих уравнений:

$$T_{\text{в}}\frac{dp_{\text{в}}}{dt} + p_{\text{в}} = u_{\text{в}};$$
$$T_{\text{н}}\frac{dp_{\text{н}}}{dt} + p_{\text{н}} = p_{\text{в}};$$
$$N_{\text{т}} = K_{\text{в}}p_{\text{в}} + (1 - K_{\text{в}})p_{\text{н}},$$

где

$T_{\text{в}}$ — постоянная времени ЦВД ($T_{\text{в}} = 0{,}4$),

$p_{\text{в}}$ — давление пара в ЦВД,

$u_{\text{в}}$ — управляющее воздействие клапана ЦВД,

$T_{\text{н}}$ — постоянная времени СПП и ЦНД ($T_{\text{н}} = 6$),

$p_{\text{н}}$ — давление пара в ЦНД,

$N_{\text{т}}$ — суммарная мощность турбины, создаваемая ЦВД и ЦНД,

$K_{\text{в}}$ — коэффициент мощности ЦВД ($K_{\text{в}} = 0{,}4$).

Уравнение ротора турбоустановки можно представить в упрощенном виде:

$$T_P\frac{df}{dt} = N_{\text{т}} - K_{\text{н}}N_{\text{г}},$$

T_P — постоянная времени ротора ($T_P = 12$),

f — частота вращения ротора,

$N_{\text{г}}$ — мощность генератора,

$K_{\text{н}}$ — коэффициент нагрузки ($K_{\text{н}} = 1$).

Математическая модель генератора определяется следующим уравнением:

$$T_э \frac{dN_г}{dt} + N_г = f,$$

где $T_э$ — постоянная времени энергосистемы ($T_э = 10$).

Приведенные выше модели проточной части, ротора и генератора должны обеспечить удовлетворительное воспроизведение динамических характеристик паровой турбины.

На рисунке 4.3.2 представлена общая структурная схема математической модели турбины для исследования переходного процесса при изменениях управляющего воздействия.

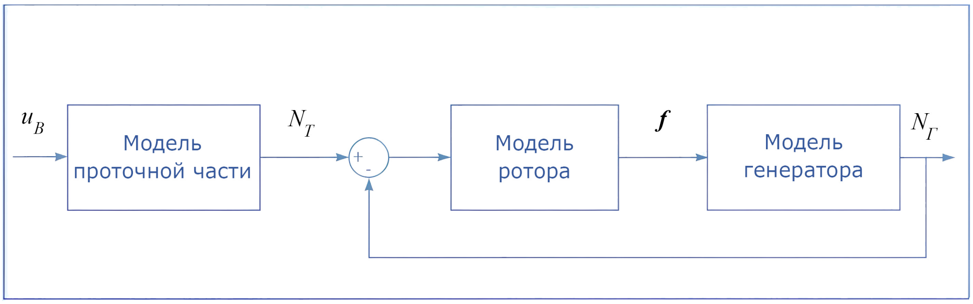

Рис. 4.3.2. Общая структурная схема математической модели турбины

Описание регулятора мощности

Регулятор мощности паровой турбины представляет собой пропорционально-интегральный регулятор, который предназначен для корректировки электрической нагрузки генератора в соответствии с заданной величиной в определенном диапазоне нагрузок.

С помощью окна управления системы верхнего уровня оператор самостоятельно задает минимальную ($Nmin$ = 300 МВт) и максимальную ($Nmax$ = 1 250 МВт) границы изменения задания регулятору мощности турбины, а также допустимую скорость изменения задания мощности ($V_{нагр}$ = 12 МВт/мин). В рамках задачи финального проекта предлагается задавать указанные выше параметры в виде постоянных значений (констант).

После завершения начального нагружения турбогенераторной установки и по команде оператора АЭС регулятор мощности безударно включается в работу и выводит мощность на заданное программой нагружения значение ($N_у$ = 1 152 МВт). При этом на вход регулятора мощности поступает нулевое рассогласование, сформированное в виде разности между заданным оператором значением и равным ему по величине текущим значением мощности.

В начальном состоянии системы:

- регулятор реактора находится в режиме поддержания давления в ГПК,
- активен турбинный регулятор мощности,
- заданное значение мощности соответствует допустимым границам рабочего диапазона изменения мощности,

- показания датчика мощности не отличаются от заданного значения,
- выходное управляющее воздействие регулятора отсутствует.

Через некоторое время оператор устанавливает величину конечного задания мощности, равной номинальному значения мощности ($N_\text{у} = N_\text{ном}$). С этого момента на вход регулятора мощности начинает поступать ненулевое рассогласование, сформированное в виде разности между новым заданным значением (с учетом допустимой скорости изменения задания мощности $V_\text{нагр}$) и текущим значением мощности.

Уравнение регулятора мощности можно представить в виде:

$$(N_\text{у} - N)K_\text{П} + K_\text{И}\int_0^t (N\text{у} - N)dt = u_N,$$

N — мощность по показаниям датчика, $N_\text{у}$ — заданная оператором мощность, $K_\text{П}$ — коэффициент пропорциональности, $K_\text{И}$ — коэффициент интегрирования, u_N — управляющее воздействие.

Обратная связь системы управления содержит датчик мощности, динамика которого описывается линейным уравнением с запаздыванием:

$$y_i(t) = x_i(t - T_\text{Д}),$$

где $T_\text{Д}$ — время задержки сигнала датчика мощности.

На рисунке 4.3.3 представлена общая структурная схема математической модели системы с замкнутым контуром регулирования мощности для исследования переходных процессов при линейных изменениях задания мощности.

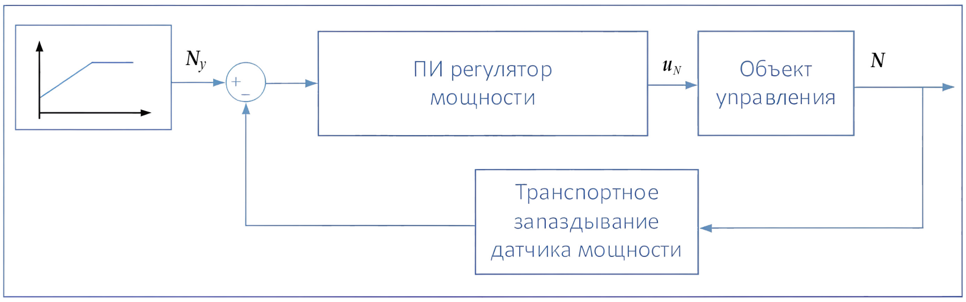

Рис. 4.3.3. Общая структурная схема системы регулирования мощности

С помощью указанных выше параметров системы посредством функционально-блочного программирования платформы SimInTech:

- произведите настройку регулятора;
- изучите влияние значений коэффициентов ПИ-регулятора на вид переходного процесса с помощью моделирования работы системы автоматического регулирования при различных значениях коэффициентов регулятора мощности турбины.

С описанием типовых законов регулирования и настройкой их параметров можно ознакомиться в справочной системе SimInTech: Лабораторные работы по вузам / АЧИИ / Моделирование системы автоматического регулирования с ПИД-регулятором (`https://help.simintech.ru/#4_nachalo_raboty/laboratornye_raboty_organizacii/ACHII/DAT_modelirovanie_sistemy_avtomaticheskogo_regulirovaniya_s_pid_regulyatorom.html`).

После успешной настройки выполните расчет времени установления системой управления номинального значения мощности в автоматическом режиме. В ответе укажите максимальное значение мощности (с точностью до 0,001 МВт), полученное в процессе регулирования. Укажите, через какое время (с точностью до 0,1 с) после изменения задания оператором мощность достигнет впервые номинального значения 1 200 МВт. Определите, через какое время значение вернется в трубку установившегося номинального значения $1\,200 \pm 0{,}1$ МВт.

Описание видеокадра оператора

Видеокадр — это проект в виде интерактивной и анимированной структурной схемы, позволяющий при моделировании оказывать воздействие на алгоритм или модель и наблюдать результаты работы.

Видеокадры в SimInTech — это стандартные проекты SimInTech, а также блоки **Субмодель** из вкладки/библиотеки **Субструктуры** в составе проектов. Они наполнены виртуальными приборами, предназначенными для отображения результатов моделирования (**Bar** — линейный прибор, **FillCircl**e — залитый эллипс, **TextLabel** — текст с функцией отображения). Кроме того, используются для управления (**Button** — кнопка, **Edit** — редактор с функцией ввода), поскольку процесс расчета модели должен включать элементы интерактивности для реализации окна оператора турбоустановки в рамках системы верхнего блочного уровня (СВБУ) АЭС. Виртуальные средства отображения и органы управления разрабатываются на основе графических примитивов SimInTech с использованием системы анимации.

Графические примитивы могут применяться для компоновки наглядных иллюстраций и сопровождающих текстов, поясняющих работу представленных в проекте SimInTech моделей и алгоритмов. Помимо статичных изображений и текстов с их помощью создается наглядная динамическая визуализация процесса расчета модели или алгоритма — виртуальные индикационные приборы, отображающие тем или иным способом значения рассчитываемых величин.

Эта возможность реализуется в SimInTech с помощью средств технической анимации, которые устанавливают связь между значениями рассчитываемых величин и свойствами графических примитивов. Для управления процессом расчета с помощью анимации можно создавать виртуальные органы управления на основе графических примитивов. В итоге внутри проекта располагаются видеокадры, взаимодействующие с его сигналами.

Ограничения на количество объектов в видеокадре нет, т. к. это стандартный проект SimInTech, однако следует учитывать возможности компьютера, на котором будет запускаться проект с видеокадрами.

Наполнение видеокадра определяется требованиями задачи управления оператором в режиме регулирования мощности турбогенератора путем формирования конечного задания мощности. При этом необходимо учитывать имеющиеся ограничения по турбине (допустимые граничные значения задания мощности и скорость ее изменения).

Видеокадр обычно повторяет внешний вид показывающих приборов диспетчерских щитов, поэтому общее окно управления турбогенератора в режиме регулирования мощности должно содержать несколько табло (показывающих приборов, панелей, индикаторов) для отображения следующих аналоговых или двоичных сигналов:

- конечное задание мощности;
- текущее задание мощности;
- темп изменения задания мощности;
- превышение рассогласования регулятора мощности на 20 МВт;
- выход за диапазон значения задания максимальной мощности;
- выход за диапазон значения задания минимальной мощности;
- текущая мощность и другие параметры.

Управление заданием мощности в ручном режиме с панели системы верхнего блочного уровня (СВБУ) допускается только после завершения начального нагружения до номинального значения мощности. Поэтому помимо показывающих приборов видеокадр должен содержать отдельное окно управления заданием мощности с помощью ручных команд. Оно включает следующие элементы управления для возможности изменения задания мощности в ручном режиме:

- кнопка переключения режима задания мощности;
- кнопки уменьшения и увеличения задания мощности;
- окно ввода задания мощности.

Для лучшего визуального восприятия стоит показывать оператору информацию о том, что органы ручного управления в данный момент включены/отключены и разрешено/запрещено управлять заданием мощности. Кроме того, необходимо разместить график переходного процесса в окне видеокадра.

На графике должны отображаться два основных параметра системы регулирования: значение текущей и заданной мощности. С описанием указанного графического примитива можно ознакомиться в справочной системе SimInTech: Графический редактор / Графические примитивы / График (Plot) (`https://help.simintech.ru/#9_grafika_i_animaciya/2_graficheskie_primitivy/Plot.html`).

Источниками и приемниками используемых на видеокадре значений могут служить как сигналы, полученные из функционально-блочной схемы проекта, в котором создаются основные модели системы регулирования турбины, так и сигналы скрипта, в котором программируется поведение основных элементов видеокадра.

С помощью блоков **Субмодель** из вкладки/библиотеки **Субструктуры** можно разделить модель турбины и основной видеокадр, а также создавать разветвленные иерархически организованные наборы частей модели турбины и видеокадров.

Для корректного отображения анимации схемное окно проекта видеокадра должно быть включено в режим «Индикация» по нажатию на кнопку  /  на панели инструментов.

Для того чтобы иметь возможность визуализировать значения, генерируемые в расчетной модели турбины, нужно создать под хранение этих значений необходимое количество сигналов проекта. Для использования в модели значений состояния элементов видеокадра также необходимо создать определенный набор сигналов.

Сигналы проекта создаются и конфигурируются в окне редактора сигналов, вызываемом через меню в Главном Окне: **Сервис** → **Сигналы**. Для дополнения блок-схемы модели блоками записи значений в сигналы проекта используются блоки 

«Запись в список сигналов» вкладки/библиотеки **Сигналы** в главном окне проекта.

Для дополнения блок-схемы модели блоками чтения значений из сигналов проекта используются блоки  «Чтение из списка сигналов» вкладки/библиотеки **Сигналы** в главном окне проекта. Для указания сигнала, в который (или из которого) должно записываться (или считываться), подключаемое к блоку значение, имя этого сигнала нужно вписать в поле свойств **Имена сигналов** (вкладка **Свойства**, столбец **Значение**) данного блока. Оно вызывается по двойному клику левой кнопкой мыши на этом блоке.

Скрипт проекта видеокадра создается с помощью нажатия на кнопку **Скрипт** панели проекта и дальнейшего формирования кода программы на специализированном языке программирования высокого уровня SimInTech, который предназначен:

- для создания программ, описывающих функционирование типового блока **Язык программирования**;
- для задания глобальных констант и переменных проекта;
- для написания типовых подпрограмм;
- для организации параметрических моделей оборудования;
- для создания программ, которые могут производить по мере расчета манипуляции с объектами схемы;
- для программирования видеокадров.

Язык программирования содержит встроенные ключевые слова, константы, декларации, операторы и функции/процедуры. Его главная особенность: текст программы предназначен для исполнения на каждом расчетном шаге при моделировании схемы.

Описание имитации отключения датчика мощности для проверки защит турбины от недопустимого разгона

Участникам необходимо создать отдельный видеокадр, который позволяет выполнять проверку срабатывания противоразгонной защиты (ПРЗ) по превышению частоты вращения турбины больше 1 650 об/мин с возможностью подключения дифференциальной защиты (ДЗ) по превышению ускорения.

Разгон можно выполнить с помощью имитации отключения обратной связи по датчику мощности. Окно имитации отключения датчика мощности должно включать следующие элементы для проведения указанных испытаний:

- кнопка имитации отключения датчика мощности;
- текущее значение частоты вращения турбины;
- превышение уставки ПРЗ по частоте вращения, равной 1 650 об/мин;
- состояние срабатывания ПРЗ;
- кнопка включения ДЗ;
- текущее значение ускорения турбины;
- состояние срабатывания ДЗ.

Подключение алгоритма ДЗ к проекту SimInTech можно выполнить любым удобным способом, например:

- воспользоваться готовым блоком **Ломаная статическая характеристика** из вкладки/библиотеки **Интерполяция** для добавления на общую схему модели системы;

- добавить код ДЗ на языке SimInTech с помощью добавления скрипта в отдельном блоке **Субмодель** или в самом проекте.

При подключении алгоритма ДЗ к проекту SimInTech необходимо произвести умножение выходного сигнала блока на коэффициент 1 000.

В рамках задачи финального проекта целью имитации отключения датчика мощности является сравнение результатов работы двух независимых защит турбины (ПРЗ и ДЗ). Испытания указанных защит турбины необходимо проводить только после завершения начального нагружения до номинального значения мощности. При этом следует продемонстрировать работу системы регулирования мощности турбины с выбранными параметрами ПИ-регулятора при проверке срабатывания этих защит.

Управляющий двоичный сигнал после нажатия на кнопку имитации отключения датчика мощности должен автоматически сниматься через 0,8 с. Это позволит максимально приблизить условия проведения эксперимента для сравнения результатов работы двух защит.

После проведения испытаний необходимо зафиксировать преимущества работы одной из защит.

4.3.6. Система оценивания

Задача командного тура оценивается максимум в 100 баллов по собственной шкале.

Таблица 4.3.2. Критерии оценивания

Наименование критерия	**Количество баллов**
Критерий 0. Этика поведения команды	**10**
Критерий 1. Адекватность разработанной математической модели	**10**
1.1 Математические выражения для объекта управления составлены верно	4
1.2 Математические выражения для регулятора составлены верно	2
1.3 Математическое описание датчика представлено корректно	2
1.4 Параметры разработанной модели выбраны корректно	2
Критерий 2. Корректность реализации модели в программном обеспечении	**20**
2.1 Приведено исчерпывающее описание переменных и параметров модели в базе данных	5
2.2 Функционально-блочная схема реализована корректно	5
2.3 Настройка элементов блочной схемы проведена корректно	5
2.4 Общие настройки параметров расчета модели заданы корректно	5
Критерий 3. Информативность и эргономичность графического интерфейса	**20**

Наименование критерия	**Количество баллов**
3.1 Предусмотрена выдача всей необходимой информации для отражения режима работы турбины в режиме реального времени	3
3.2 Представлены все необходимые элементы управления для реализации заданной логики управления турбиной	3
3.3 Реализованы защиты и их сигнализация	3
3.4 Корректно отображена графическая модель турбины	3
3.5 График переходного процесса и коэффициенты регулятора корректно отображаются в окне видеокадра	3
3.6 Эргономичность интерфейса (выбранная цветовая гамма, размеры и размещение элементов интерфейса не вызывают утомляющего эффекта у оператора; расположение элементов управления интуитивно понятно)	5
Критерий 4. Соответствие разработки предъявляемым требованиям и исходным данным	**20**
4.1 Разработанная модель системы управления позволяет исследовать динамику переходных процессов	3
4.2 Разработанная модель позволяет проводить испытания системы регулирования и защит турбины при отказах датчиков мощности в рамках заданного режима работы АЭС	3
4.3 Закон управления удовлетворяет минимальным требованиям, предъявляемым к качеству переходного процесса	3
4.4 Реализованы требования по повышению надежности паротурбинной установки, увеличении ее КПД, предупреждению аварийных ситуаций	3
4.5 Выполнена самостоятельная настройка регулятора мощности турбины и предоставлены результаты улучшения качества переходного процесса	3
4.6 Разработанная система не противоречит технологическому процессу	4
Критерий 5. Качество представленных результатов	**20**
5.1 Продемонстрирован сценарий работы контура управления в автоматическом режиме в соответствии с заданными условиями	4
5.2 Подготовлен и представлен отчет, включающий описание переменных и параметров в проекте SimInTech, программный код в проекте SimInTech, скрины функционально-блочной схемы, графического интерфейса, краткие выводы по динамическим характеристикам разработанной модели объекта управления, по эффективности различных алгоритмов защит турбины в результате испытаний при имитации отказов оборудования	10
5.3 Представленный проект работоспособен	3
5.4 Достаточность представленного материала для понимания логики работы модели	3
Итого	100

4.3.7. Решение задачи

Описание переменных и параметров

№	Имя	Название	Тип данных	Режим	Формула	Значение	Способ расчёта
1	startN	Стартовое значение мощности	Вещественное	Вход	1152	1152	**Константа**
2	targetN	Целевое значение мощности	Вещественное	Вход	1200	1200	Переменная
3	currentN	Текущее значение мощности	Вещественное	Вход	1152	1152.070405	Переменная
4	Nmin	Минимальное значение мощности	Вещественное	Вход	300	300	**Константа**
5	Nmax	Максимальное значение мощности	Вещественное	Вход	1250	1250	**Константа**
6	rtw	Готовность работы ручного управления	Двоичное	Вход	1	☐Нет	Переменная
7	old_time	Время, когда в последний раз мощность устанавливалась вручную	Вещественное	Вход	0	0	Переменная
8	turnOffDetector	Имитация отключения датчика	Двоичное	Вход	0	☐Нет	Переменная
9	ciclesCounter	Кол-во циклов	Целое	Вход	0	0	Переменная
10	lastN	Последнее значение мощности	Вещественное	Вход	1152	1152	Переменная
11	rotorF	Частота вращения ротора	Вещественное	Вход	1152	1152.385223	Переменная
12	turnOff	Выключение турбины в соответствии с ПРЗ	Двоичное	Вход	0	☐Нет	Переменная
13	currentTargetN	Целеворе значение текущей мощности	Вещественное	Вход	1152	1153.451	Переменная
14	setTime	Время, когда в последний раз мощность устанавливалась вручную	Вещественное	Вход	0	0	Переменная
15	setCurrentTargetN	Последнее заданное вручную значение мощности	Вещественное	Вход	1152	1152	Переменная
16	rotorAcc	Ускорение ротора	Вещественное	Вход		0.1255603831	Переменная
17	turnOnDiffProt	Переменная включения дифференциальной защиты	Двоичное	Вход	1	☐Нет	Переменная

Рис. 4.3.4. Переменные, используемые в проекте

В программе задано несколько переменных констант: `Nmin`, `Nmax`, `startN`, они не меняются по ходу программы, однако перед каждым запуском их можно отредактировать.

Кроме того, система управления турбиной обращается к переменным: `startrN`, `targetN`, `currentN`, `lastN`, `rotorF`, `rotorAcc`, `turnOnDiffProt` и `setCurrentTargetN`.

Система имитации отключения датчика обращается к переменным: `ciclesCount`, `turnOffDetector`.

После этого система графического интерфейса обращается ко всем необходимым для оператора переменным для их визуализации в видеокадре.

Структурная схема

Рис. 4.3.5. Общий план

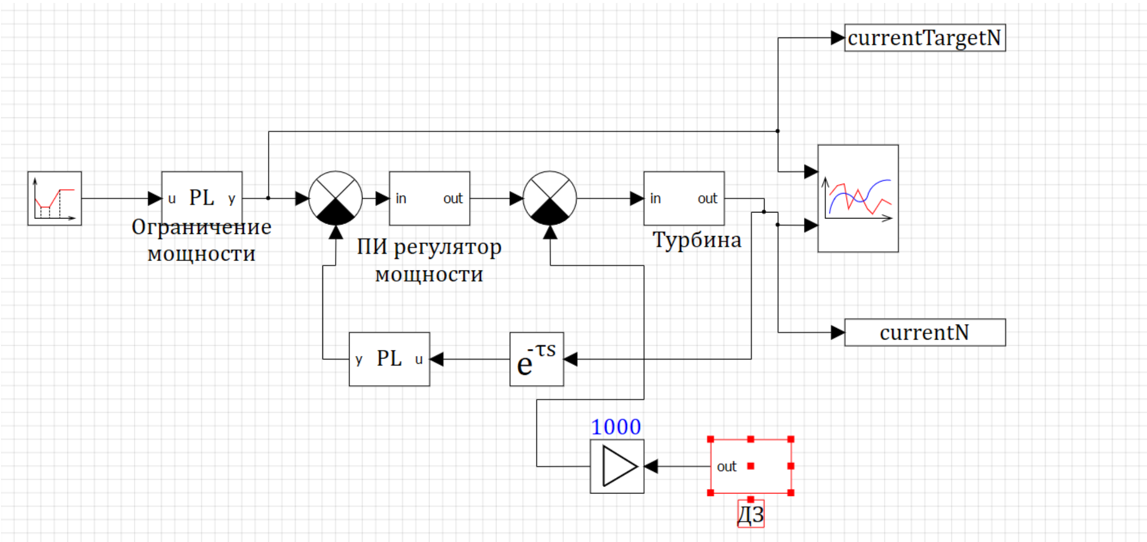

Рис. 4.3.6. Система управления турбиной

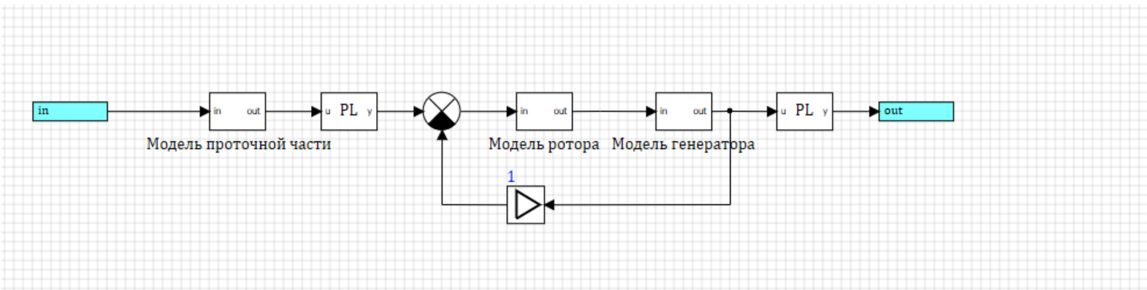

Рис. 4.3.7. Модель турбины

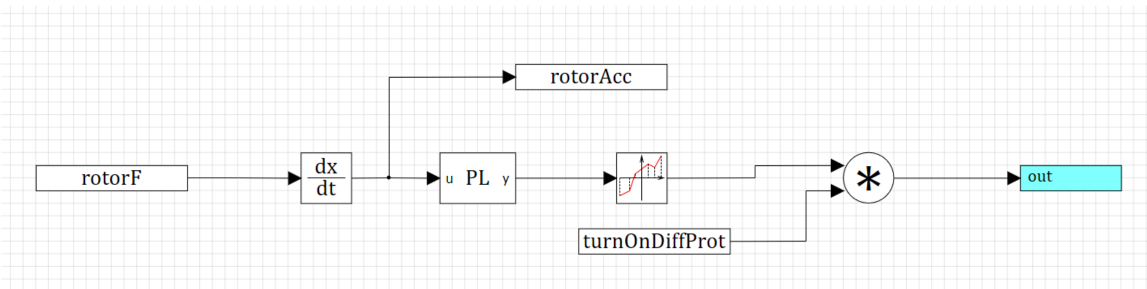

Рис. 4.3.8. Схема дифференциальной защиты

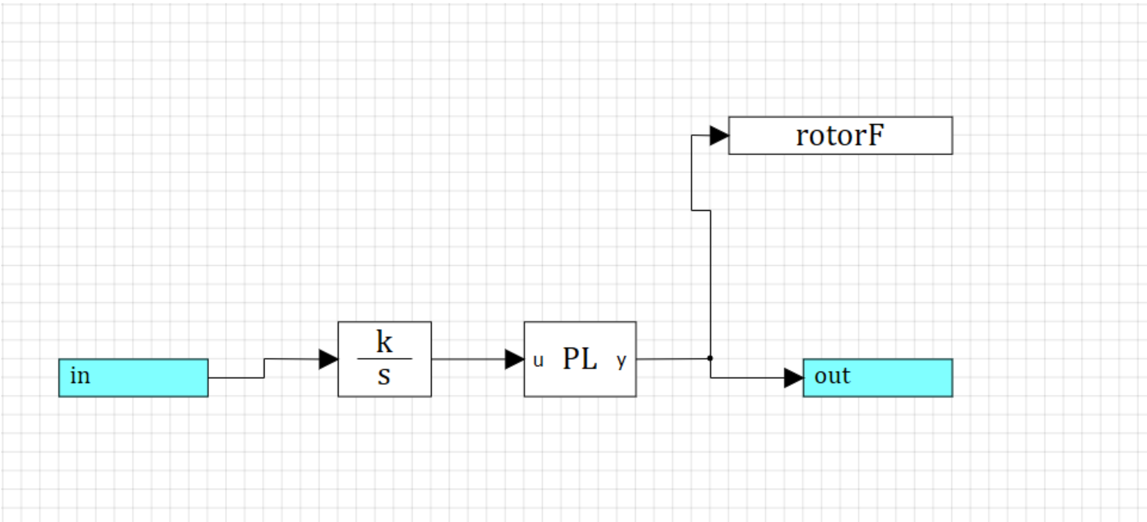

Рис. 4.3.9. Модель ротора

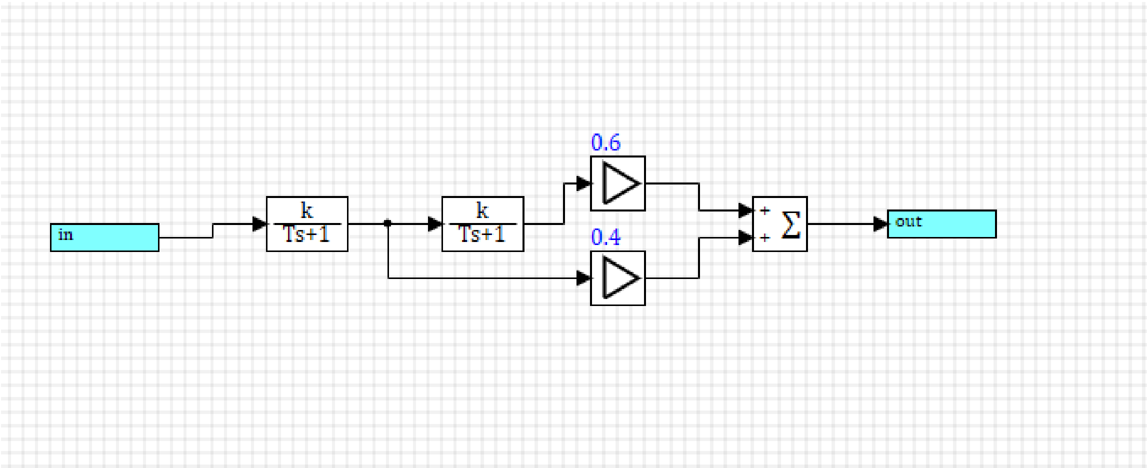

Рис. 4.3.10. Модель проточной части

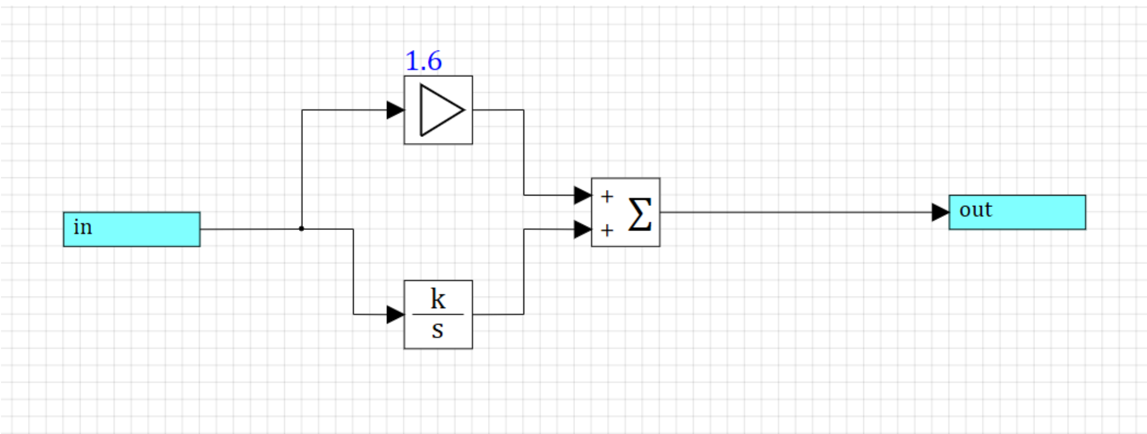

Рис. 4.3.11. ПИ-регулятор мощности

Графический интерфейс (видеокадр)

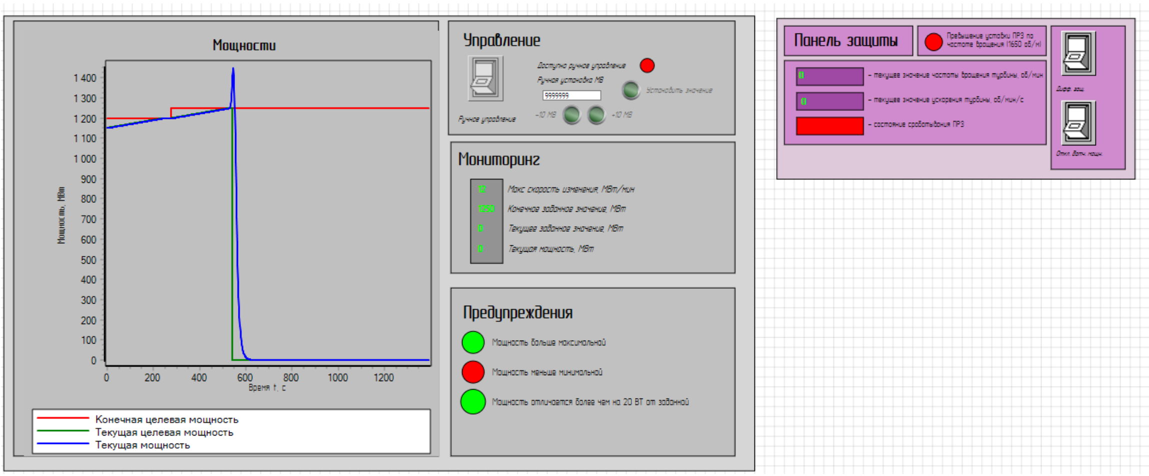

Рис. 4.3.12. Основной интерфейс управления турбиной

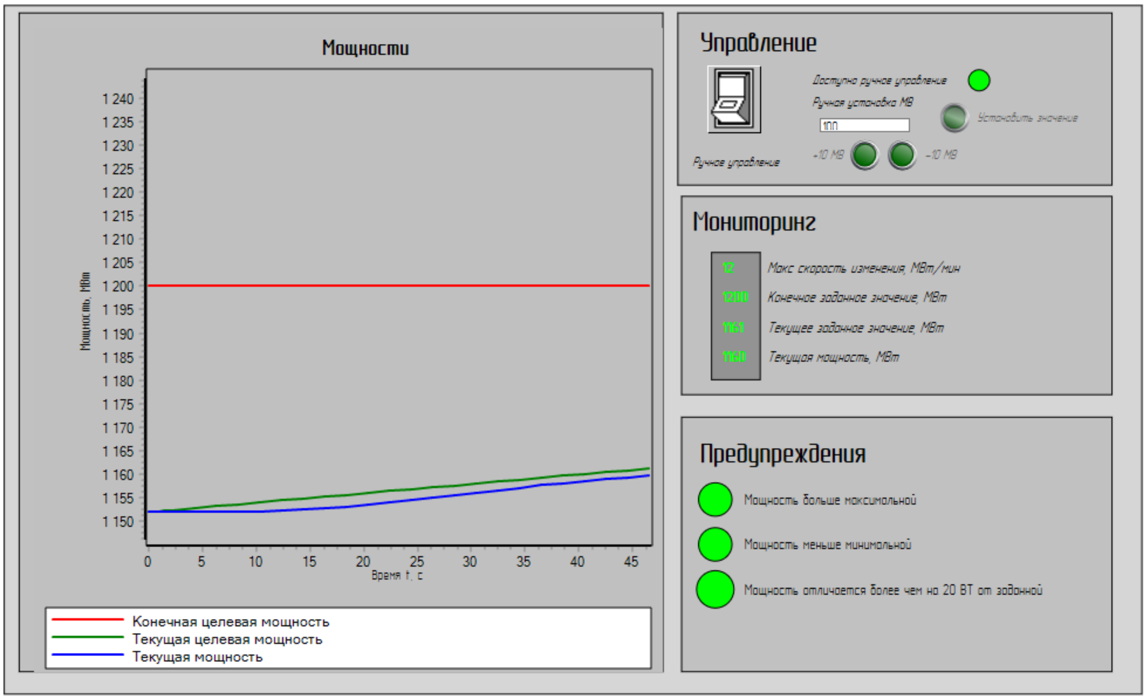

Рис. 4.3.13. Панель отслеживания и регулирования мощности турбины

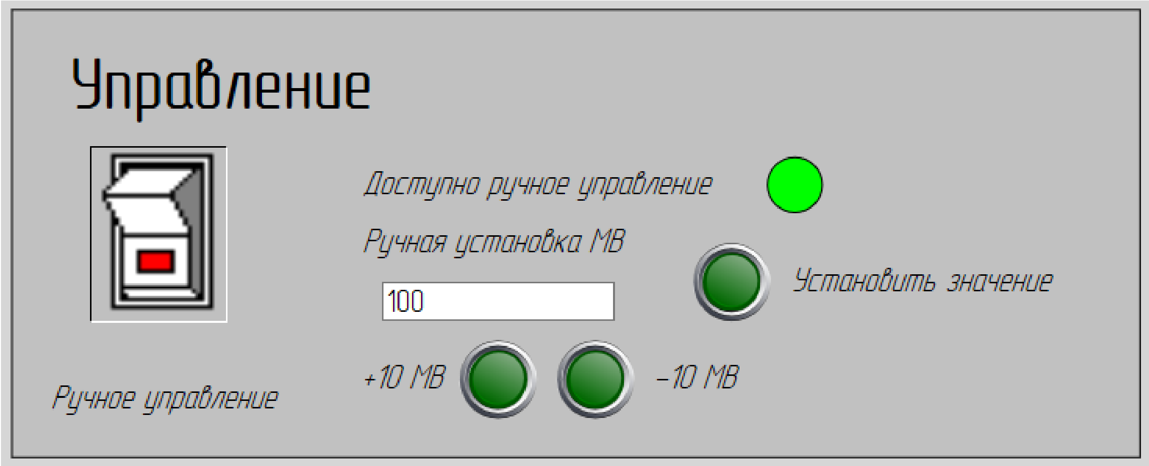

Рис. 4.3.14. Панель переключения на ручное управление. Ручное управление доступно и включено

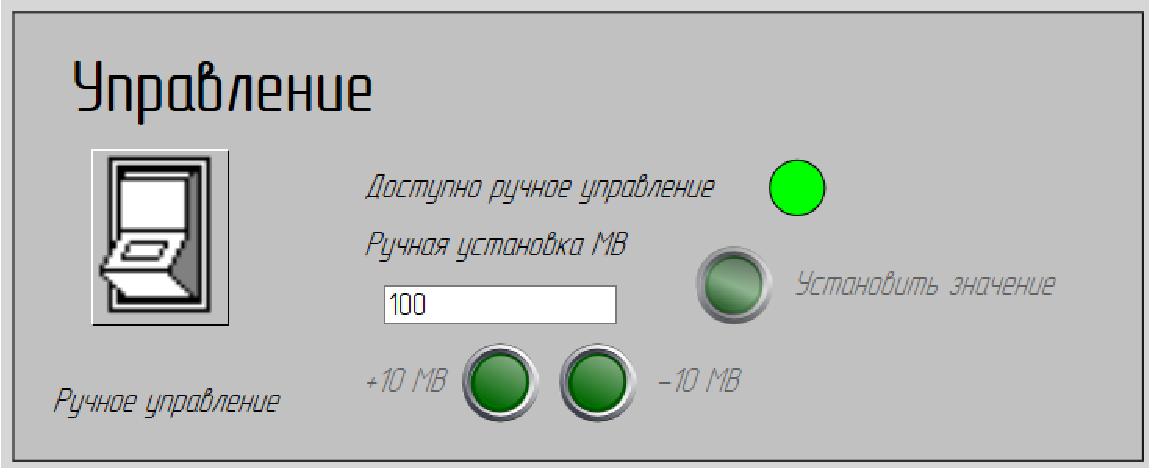

Рис. 4.3.15. Ручное управление выключено, но доступно

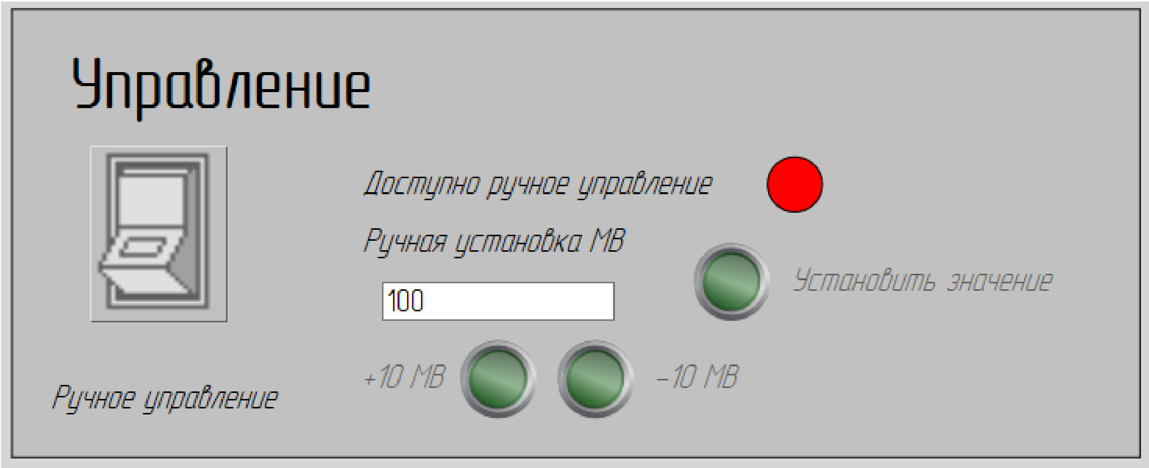

Рис. 4.3.16. Ручное управление недоступно

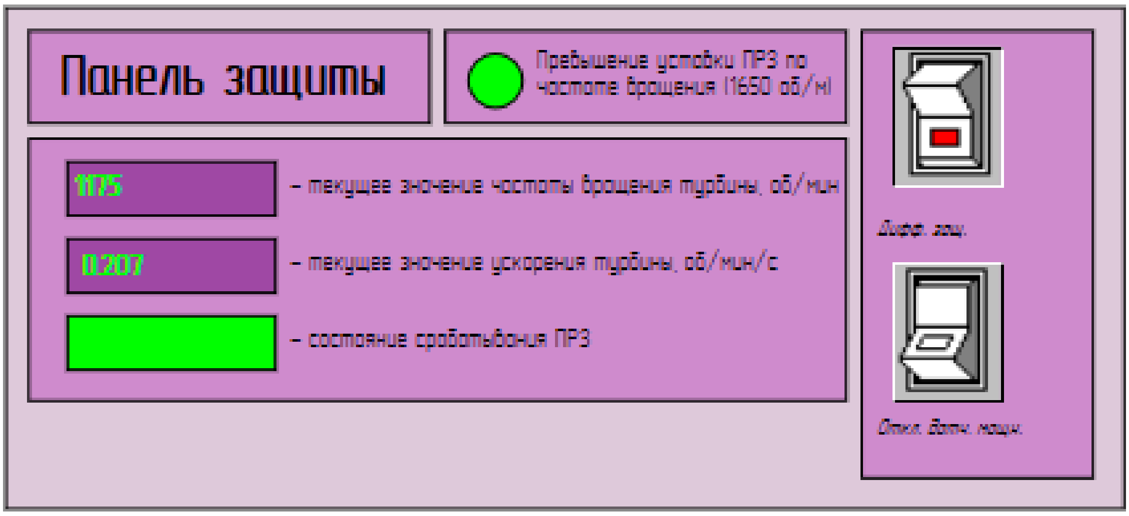

Рис. 4.3.17. Панель противоразгонной защиты

Рассмотрим полную панель управления и отслеживания состояния паровой турбины. Интерфейс позволяет оператору быстро регулировать и изменять состояние турбины (рис. 4.3.12). Более детальная панель (рис. 4.3.13) — слева оператор может увидеть график, позволяющий контролировать настоящую мощность турбины, текущую целевую мощность и конечную целевую мощность.

В правом нижнем углу установлены два блока для мониторинга состояния турбины в реальном времени, а также панель предупреждений — если указанный параметр превышает безопасное значение индикатор загорается красным цветом.

Панель переключения на ручное управление (рис. 4.3.14). В данный момент ручное управление доступно и включено: оператор может самостоятельно установить значение МВ. Ситуация, когда ручное управление не включено, но доступно (рис. 4.3.15). Мощность не достигла номинального значения МВ (рис. 4.3.16), поэтому в данный момент ручное управление недоступно, что показывается красным цветом индикатора.

В нижнем правом углу панели противоразгонной защиты (рис. 4.3.17) оператор может симулировать отключение датчика мощности для разгона турбины. Выше представлена кнопка включения дифференциальной защиты.

Основная часть панели позволяет отслеживать основные состояния турбины. Выше представлен индикатор превышения установки ПРЗ по частоте вращения: если частота превышает установку, индикатор загорается красным цветом.

Программный код проекта

Python

```
ciclesCounter = ciclesCounter + 1 // Счётчик циклов для точного
    ↪ контроля времени

// Логика ручного управления

if rtw = 0   // Если ручное управление недоступно, то весь интерфейс
    ↪ взаимодействия с ним сделать полупрозрачным
```

```
  then begin
  ctrl_switch.Down = 0
  ctrl_switch.Opacity = 0.5
  ctrl_led.Color = rgbtocolor(255, 0, 0)
  ctrl_btn_up.Opacity = 0.5
  ctrl_btn_down.Opacity = 0.5
  ctrl_btn_set.Opacity = 0.5
  end
 else begin    // Если ручное управление доступно, то сделать видимым тумблер его включения и переключить индикационный светодиод
  ctrl_switch.Opacity = 1
  ctrl_led.Color = rgbtocolor(0, 255, 0)
  ctrl_btn_up.Opacity = 1
  ctrl_btn_down.Opacity = 1
  ctrl_btn_set.Opacity = 1

  hand_control = ctrl_switch.Value

  if hand_control = 1 then begin   // Если ручное управление доступно и включен тумблер ручного управления, то кнопки ручного управления сделать видимыми
          ctrl_text_2.Opacity = 1
          ctrl_text_3.Opacity = 1
          ctrl_text_4.Opacity = 1
          ctrl_btn_set.Opacity = 1

          if (ctrl_btn_set.state = 1) then begin // Если установлена мощность через текстовое меню, передать её в переменную targetN
          targetN = ctrl_text_editor.value
          setTime = time
          setCurrentTargetN = currentTargetN
          end

          if (ctrl_btn_down.state = 1) and (time - old_time > 0.4) then begin // Если нажата кнопка уменьшения мощности, мощность понижается
          targetN = targetN - 10
          setTime = time
          old_time = time
          setCurrentTargetN = currentTargetN
          end

          if (ctrl_btn_up.state = 1) and (time - old_time > 0.4) then begin   // Если нажата кнопка увеличения мощности, мощность увеличивается
          targetN = targetN + 10
          setTime = time
          old_time = time
          setCurrentTargetN = currentTargetN
          end

  end
  else begin  // Если ручное управление доступно но не включен тумблер ручного управления, то кнопки ручного управления сделать полупрозрачными
          ctrl_text_2.Opacity = 0.5
          ctrl_text_3.Opacity = 0.5
          ctrl_text_4.Opacity = 0.5
          ctrl_btn_set.Opacity = 0.5
```

```
 end
end

// Логика отключения считывания датчика на 0.8 секунд

if turnOnDiffProtSwitch.Value then turnOnDiffProt = True  // Если тумблер включения дифференциальной защиты включен, переключить переменную, отвечающую за дифференциальную защиту
else turnOnDiffProt = False

if not turnOffDetector then   // Если тумблер включения имитации ошибки датчика, то датчик и иумблер выключаются на 0.8 с
begin
if turnOffDetectorSwitch.Down then turnOffDetector = True
ciclesCounter = 0
end
else if ciclesCounter >= 800 then
begin
turnOffDetector = False
turnOffDetectorSwitch.Down = False
end

// Логика индикаторов

if currentN > Nmax then led_high.color = rgbtocolor(255, 0, 0)   // Если мощность турбины больше максимальной, то включить индикатор
else led_high.color = rgbtocolor(0, 255, 0) ;
if targetN > Nmax then targetN = Nmax;

if currentN < Nmin then led_low.color = rgbtocolor(255, 0, 0)   // Если мощность турбины меньше минимальной, то включить индикатор
else led_low.color = rgbtocolor(0, 255, 0);
if targetN < Nmin then targetN = Nmin;

if round(abs(currentN - currentTargetN)) > 20 then led_error.color = rgbtocolor(255, 0, 0) // Если текущая мощность отличается от заданной на 20 МВат, то включить индикатор
else led_error.color = rgbtocolor(0, 255, 0) ;


if turnOff then VPSIndicator.color = rgbtocolor(255, 0, 0)  // Переключения индикатора ПРЗ
else VPSIndicator.color = rgbtocolor(0, 255, 0)

if rotorF > 1650 then fLimitIndicator.color = rgbtocolor(255, 0, 0) // Переключения индикатора выхода за 1650 об/мин
```

Выводы

Сравнение двух методов защиты турбины показало, что дифференциальная защита является более безопасным и продвинутым способом. Турбина в нештатных ситуациях с подключенной дифференциальной защитой способна замедлить набор частоты вращения и вернуть ее в контролируемый диапазон, не доходя до критической частоты.

Несмотря на преимущество дифференциальной защиты, на реакторных установках важно присутствие обеих систем, где противоразгонная защита срабатывает в критических случаях, когда по определенным причинам дифференциальная защита неспособна стабилизировать турбину.

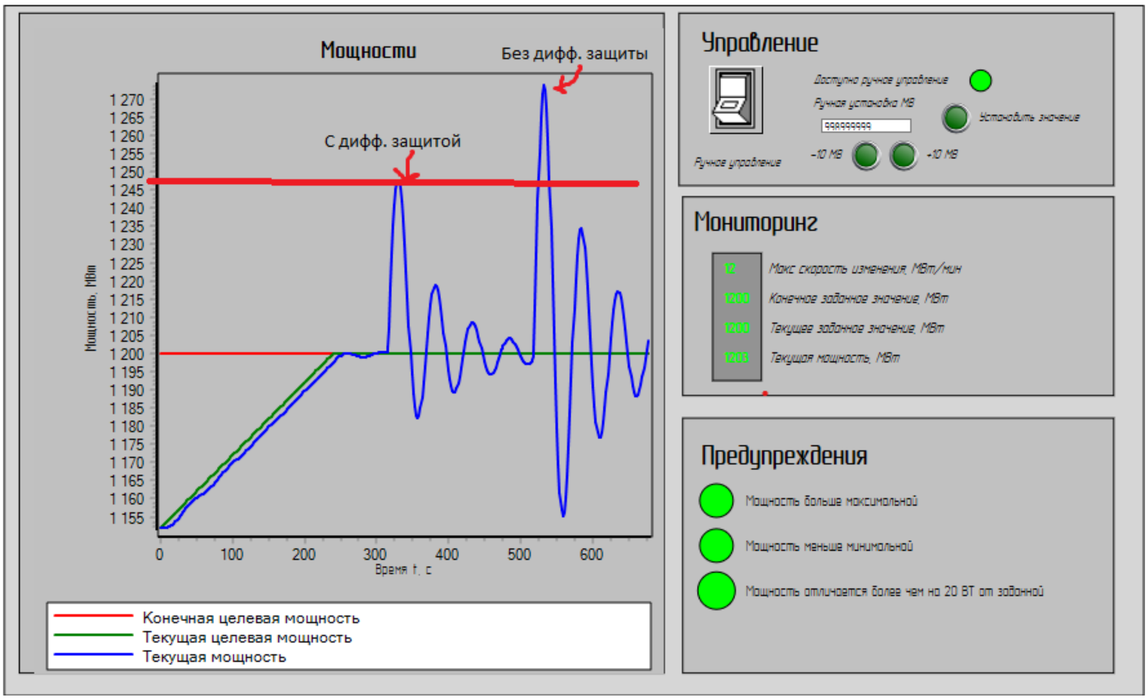

Рис. 4.3.18. График сравнения мощностей с дифференциальной защитой и без нее

Выходя за рамки задания, стоит отметить, что, помимо реализованных систем защиты, существуют также два варианта защиты турбины от разгона: использование low-pass фильтра для фильтрации значений с датчиков или, при возвращении датчиком нулевых значений, ориентировка на последнее полученное с датчика ненулевое показание.

Ответ:

- 392,2 с — устоялось номинальное значение;
- 257,8 с — время первого достижения номинального значения;
- 1200,37 МВт — максимальное значение мощности.

4.3.8. Материалы для подготовки

1. ООО «СимИнтех» [Электронный ресурс]. — Режим доступа: https://simintech.ru/, свободный.
2. OpenEdu — Национальная платформа открытого образования [Электронный ресурс]. — Режим доступа: https://openedu.ru/, свободный. (Курсы: «Проект реактора ВВЭР», «Основы энергетических ядерных технологий», «Язык программирования C++», «Дифференциальные уравнения»).
3. Stepik — Образовательная платформа [Электронный ресурс]. — Режим доступа: https://stepik.org/, свободный. (Курсы: «Ядерная физика — просто!», «Практикум по программированию», «Введение в программирование (C++)»).

4. Харитонов В. В. Энергетика. Технико-экономические основы: учебное пособие. — М.: МИФИ, 2007.
5. Наумов В. И. Физические основы безопасности ядерных реакторов: учебное пособие. — 2-е изд. — М.: НИЯУ МИФИ, 2013.
6. Семенов В. К. Кинетика и регулирование ядерных реакторов: учебное пособие. — Иваново: Ивановский государственный энергетический университет, 2009.
7. Лескин С. Т., Шелегов А. С., Слободчук В. И. Физические особенности и конструкция реактора ВВЭР-1000: учебное пособие. — М.: НИЯУ МИФИ, 2011. — 116 с.
8. Машиностроение ядерной техники. В 2 кн. Кн. 1, 2 / ред.-сост. В. И. Солонин; отв. ред. К. С. Колесников. — М., 2005. — (Т. IV-25).

5. Критерии определения победителей и призеров

Первый отборочный этап

В первом отборочном этапе участники решали задачи предметного тура по двум предметам: физике и информатике и инженерного тура. В каждом предмете максимально можно было набрать 100 баллов, в инженерном туре 100 баллов. Для того чтобы пройти во второй этап, участники должны были набрать в сумме по обоим предметам и инженерному туру не менее 10,0 баллов, независимо от уровня.

Второй отборочный этап

Количество баллов, набранных при решении всех задач второго отборочного этапа, суммируется. Победители второго отборочного этапа должны были набрать не менее 200,0 баллов, независимо от уровня.

Заключительный этап

Индивидуальный предметный тур

- физика — максимально возможный балл за все задачи — 100 баллов;
- информатика — максимально возможный балл за все задачи — 100 баллов.

Командный инженерный тур

Команды заключительного этапа получали за командный инженерный тур от 0 до 100,00 баллов: команда, набравшая наибольшее число баллов среди других команд, становилась командой-победителем.

Все результаты команд нормировались по формуле:

$$\frac{100 \times x}{MAX},$$

где x — число баллов, набранных командой,

MAX — число баллов, максимально возможное за инженерный тур.

В заключительном этапе олимпиады индивидуальные баллы участника складываются из двух частей, каждая из которых имеет собственный вес: баллы за индивидуальное решение задач по предмету 1 (физика) с весом $K_1 = 0{,}15$, по предмету 2 (информатика) с весом $K_2 = 0{,}15$, баллы за командное решение задач инженерного тура с весом $K_3 = 0{,}7$.

Итоговый балл определяется по формуле:

$$S = K_1 \cdot S_1 + K_2 \cdot S_2 + K_3 \cdot S_3,$$

где S_1 — балл первой части заключительного этапа по физике (предметный тур) ($S_{1\text{ макс}} = 100$);

S_2 — балл первой части заключительного этапа по информатике (предметный тур) ($S_{2\text{ макс}} = 100$);

S_3 — итоговый балл инженерного командного тура ($S_{3\text{ макс}} = 100$).

Итого максимально возможный индивидуальный балл участника заключительного этапа — 100 баллов.

Критерий определения победителей и призеров

Чтобы определить победителей и призеров (независимо от класса) на основе индивидуальных результатов участников, был сформирован общий рейтинг всех участников заключительного этапа. С начала рейтинга были выбраны 8 победителей и 17 призеров (первые 25% участников рейтинга становятся победителями или призерами, из них первые 8% становятся победителями, оставшиеся — призерами).

Критерий определения победителей и призеров (независимо от уровня)

Категория	**Количество баллов**
Победители	68,55 и выше
Призеры	От 59,90 до 67,70

6. Работа наставника после НТО

Участие школьника в Олимпиаде может завершиться после любого из этапов: первого или второго отборочных, либо после заключительного этапа. В каждом случае после завершения участия наставнику необходимо провести с учениками рефлексию — обсудить полученный опыт и проанализировать, что позволило достичь успеха, а что привело к неудаче. Подробные материалы о проведении рефлексии представлены в курсе «Наставник НТО»: https://academy.sk.ru/events/310.

Наставнику важно проинформировать руководство образовательного учреждения, если его учащиеся стали финалистами, призерами и победителями. Публичное признание высоких результатов дополнительно повышает мотивацию.

В процессе рефлексии с учениками, не ставшими призерами или победителями, рекомендуется уделить особое внимание особенностям командной работы: распределению ролей, планированию работы, возникающим проблемам. Для этого могут использоваться опросники для самооценки собственной работы и взаимной оценки участниками других членов команды (P2P). Они могут выявить внутренние проблемы команды, для решения которых в план подготовки можно добавить мероприятия, направленные на ее сплочение.

Стоит рассказать, что в истории НТО было много примеров, когда не победив в первый раз, на следующий год участники показывали впечатляющие результаты, одержав победу сразу в нескольких профилях. Конечно, важно отметить, что так происходит только при учете прошлых ошибок и подготовке к Олимпиаде в течение года.

Важным фактором успешного участия в следующих сезонах НТО может стать поддержка родителей учеников. Знакомство с ними помогает наставнику продемонстрировать важность компетенций, развиваемых в процессе участия в НТО, для будущего образования и карьеры школьников. Поддержка родителей помогает мотивировать участников и позволяет выделить необходимое время на занятия в кружке.

С участниками-выпускниками наставнику рекомендуется обсудить их дальнейшее профессиональное развитие и его связь с выбранными профилями НТО. Отдельно можно обратить внимание на льготы для победителей и призеров, предлагаемые в вузах с интересующими ученика направлениями. Кроме того, ряд вузов предлагает льготы для всех финалистов НТО, а также учитывает результаты Конкурса цифровых портфолио «Талант НТО».