

НТО

МАТЕРИАЛЫ ЗАДАНИЙ

Всероссийской междисциплинарной олимпиады

школьников 8–11 класса

«Национальная технологическая олимпиада»

по профилю

«Водные робототехнические системы»

2024/25 учебный год

УДК 373.5.016:681.51

ББК 74.263.0

В62

Авторы:

И. И. Бажанский, А. А. Заузолков, О. В. Зубков, О. И. Кабанцова, К. Д. Кириченко, Н. Ю. Кузнецов, А. П. Лобов, С. А. Мун, С. А. Никифорова, А. А. Омельяненко, В. С. Сорин

В62 Всероссийская междисциплинарная олимпиада школьников 8–11 класса «Национальная технологическая олимпиада». Учебно-методическое пособие

Том 7 Водные робототехнические системы

— М.: Ассоциация участников технологических кружков, 2025. — 191 с.

ISBN 978-5-908021-06-7

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, центров молодежного и инновационного творчества и детских технопарков.

Методические материалы также могут быть полезны студентам и преподавателям направлений, относящихся к группам:

02.00.00 Компьютерные и информационные науки

03.00.00 Физика и астрономия

09.00.00 Информатика и вычислительная техника

11.00.00 Электроника, радиотехника и системы связи

15.00.00 Машиностроение

27.00.00 Управление в технических системах

ISBN 978-5-908021-06-7

УДК 373.5.016:681.51

ББК 74.263.0



9 785908 021067 >

Оглавление

| | |
|--|------------|
| 1 Введение | 5 |
| 1.1 Национальная технологическая олимпиада | 5 |
| 1.2 Водные робототехнические системы | 13 |
| 2 Первый отборочный этап | 15 |
| 2.1 Работа наставника НТО на этапе | 15 |
| 2.2 Предметный тур. Информатика | 16 |
| 2.2.1 Первая волна. Задачи 8–11 класса | 16 |
| 2.2.2 Вторая волна. Задачи 8–11 класса | 26 |
| 2.2.3 Третья волна. Задачи 8–11 класса | 36 |
| 2.2.4 Четвертая волна. Задачи 8–11 класса | 49 |
| 2.3 Предметный тур. Физика | 64 |
| 2.3.1 Первая волна. Задачи 8–9 класса | 64 |
| 2.3.2 Первая волна. Задачи 10–11 класса | 68 |
| 2.3.3 Вторая волна. Задачи 8–9 класса | 73 |
| 2.3.4 Вторая волна. Задачи 10–11 класса | 80 |
| 2.3.5 Третья волна. Задачи 8–9 класса | 85 |
| 2.3.6 Третья волна. Задачи 10–11 класса | 90 |
| 2.3.7 Четвертая волна. Задачи 8–9 класса | 94 |
| 2.3.8 Четвертая волна. Задачи 10–11 класса | 100 |
| 2.4 Инженерный тур | 105 |
| 3 Второй отборочный этап | 115 |
| 3.1 Работа наставника НТО на этапе | 115 |
| 3.2 Инженерный тур | 117 |
| 3.2.1 Индивидуальные задачи | 118 |
| 3.2.2 Командные задачи | 133 |

| | |
|--|------------|
| 4 Заключительный этап | 136 |
| 4.1 Работа наставника НТО при подготовке к этапу | 136 |
| 4.2 Предметный тур | 138 |
| 4.2.1 Информатика. 8–11 классы | 138 |
| 4.2.2 Физика. 8–9 классы | 153 |
| 4.2.3 Физика. 10–11 классы | 159 |
| 4.3 Инженерный тур | 167 |
| 4.3.1 Общая информация | 167 |
| 4.3.2 Легенда задачи | 167 |
| 4.3.3 Требования к команде и компетенциям участников | 168 |
| 4.3.4 Оборудование и программное обеспечение | 169 |
| 4.3.5 Описание задачи | 171 |
| 4.3.6 Система оценивания | 178 |
| 4.3.7 Решение задачи | 184 |
| 4.3.8 Материалы для подготовки | 188 |
| 5 Критерии определения победителей и призеров | 189 |
| 6 Работа наставника после НТО | 191 |

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 предлагаются 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 роботехническими 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: помочь 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 Кружко-

вое 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 помочь 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. Водные робототехнические системы

Цель профиля Водные робототехнические системы — развитие у школьников прикладных навыков в сфере морской робототехники через знакомство с технологиями машинного зрения, основами теории автоматического управления, приобретение навыками проектирования подводных аппаратов и разработки технической документации.

Профиль направлен на решение проблем в области подводной робототехники, в частности, создание подводных аппаратов для практического использования в исследовательских целях, разработки полезной нагрузки и отработки алгоритмов компьютерного зрения аппарата.

В рамках заключительного этапа команде необходимо сделать микротелеуправляемый необитаемый подводный аппарат (микро-ТНПА), конструкция которого включает в себя цифровую камеру и независимый пульт управления.

С помощью аппарата-носителя MiddleROV необходимо:

1. доставить микро-ТНПА до места исследования;
2. определить тип обрастаний на корпусе морского судна;
3. провести исследование подводного шлюза;
4. определить и подсчитать количество повреждений (отверстий) в нем.

Результатом решения задачи являются собранный микро-ТНПА, функционирующий согласно техническому заданию, программное обеспечение для обработки видеопотока с микро-ТНПА для диагностики состояния морского судна, а также соответствующий комплект конструкторской документации.

Итоговое задание **командного практического тура заключительного этапа** разбито на подзадачи так, чтобы все члены команды, обладающие различными навыками, могли работать над своей частью общей задачи на протяжении всего этапа. Каждая подзадача оценивается по результату: работает или не работает, выполняет ли подводную задачу, соответствует ли критериям оценки.

Все этапы профиля направлены на то, чтобы команда приобрела дополнительные навыки и компетенции в ходе выполнения задач отборочных этапов, которые в итоге позволяют представить финальное решение технологической проблемы, рассматриваемой в рамках профиля.

В первом отборочном дистанционном этапе, который проводится индивидуально, участникам предложены задания по двум предметам — информатика и физика.

По информатике проверяются базовые основы программирования; по физике сделан упор на темы, посвященные изучению электрического тока, гидродинамики, механики, которые составляют теоретическую основу подводной робототехники.

Задачи **инженерного тура** разработаны для четырех компетенций: программирование машинного зрения, конструирование, электроника, программирование микроконтроллеров. Здесь участники готовятся ко второму и третьему этапам. В целом, задачи первого этапа выявляют наличие знаний, необходимых не только

для решения задач следующего этапа, но и финальной задачи.

Каждый участник может попробовать решить задачи по всем компетенциям или выбрать только одно направление, в котором хочет в дальнейшем развиваться в команде.

На втором отборочном этапе (дистанционном командном) предлагаются четыре типа командных задач для каждой роли:

- Программист верхнего уровня занимается программированием АНПА (автономный необитаемый подводный аппарат) в симуляторе MUR IDE. Для удобства участников работа ведется на собственной платформе профиля MUR Contest, что обеспечивает равные условия для оценки кода участников. Необходимо запрограммировать робота на автономное движение, стабилизацию и компьютерное зрение.
- Программист нижнего уровня выполняет программирование микроконтроллеров в симуляторе электронных схем SimulIDE.
- Электронщик разрабатывает электрическую схему по заявленному проекту печатной платы подводного аппарата.
- Конструктор проектирует в САПР и создает чертежи герметичного корпуса для камеры с сервоприводом по заявленными характеристиками.

Таким образом, индивидуальные задачи готовят участников к выполнению финальной задачи и направлены на развитие необходимых знаний и навыков.

В рамках командной задачи необходимо разработать техническое задание к существующему телекоммуницируемому необитаемому подводному аппарату. Для качественного выполнения работы требуется участие всех членов команды.

Для подготовки и обсуждения с участниками решения задач проводится вебинар с разработчиками, на котором каждый может задать свои вопросы. Кроме того, предоставляются материалы и уроки по программированию роботов в симуляторе, конструированию, схемотехнике и подводной робототехнике.

В итоге проведенный цикл мероприятий профиля позволяет участникам прийти к решению задачи заключительного этапа с необходимым набором навыков и компетенций, а также подготовительных материалов и разработок.

Приобретенные компетенции дают импульс к продолжению изучения подводной робототехники. Так, сформированные на профиле команды продолжают принимать участие в различных соревнованиях и конкурсах по подводной робототехнике как в роли школьников, так и в качестве студентов студентов.

Кроме того, некоторые выпускники профиля в дальнейшем становятся авторами олимпиадных задач, работают в инженерных проектах, реализуемых вузами и связанных с конструированием, программированием и электроникой.

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 Конкурс. 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 $0 \leq t \leq 10^6$.

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++

```

1 #include<bits/stdc++.h>
2 #define int long long
3 using namespace std;
4 signed main(){
5     int t;
6     cin >> t;
7     cout << ((t * (t - 1)) * (t - 2)) << endl;
8 }
```

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 $1 \leq a, b, c, d \leq 10^9$.

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++

```

1 #include<bits/stdc++.h>
2 #define int long long
3 using namespace std;
4 signed main(){
5     double a, b, c, d;
6     cin >> a >> b >> c >> d;
7     double a0 = min({a, b, c, d});
8     double a1 = min(max(a, b) / 2.0, min(a, b));
9     double a2 = min(max(c, d) / 2.0, min(c, d));
10    double ans = max({a0, a1, a2});
11    if( (int)ans == ans ){
12        int ians = ans;
13        cout << ians << endl;
14        return 0;
15    }
16    cout.precision(1);
17    cout << fixed << ans << endl;
18 }
```

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 $1 \leq n \leq 10^5$.

Во второй строке заданы $2n$ целых чисел d_i — длины всех кусков, которые получились после «тренировки» внука, где $1 \leq d_i \leq 10^9$. 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++

```

1  #include<bits/stdc++.h>
2  #define int long long
3  using namespace std;
4  signed main(){
5      int n;
6      cin >> n;
7      vector<pair<int, int> > v(2 * n);
8      for(int i = 0; i < 2 * n; i++){
9          int d;
10         cin >> d;
11         v[i] = {d, i + 1};
12     }
13     sort(v.begin(), v.end());
14     vector<pair<int, int> > ans(n);
15     for(int i = 0; i < n; i++){

```

```

16     ans[i] = {v[i].second, v[2 * n - i - 1].second};
17     if(ans[i].first > ans[i].second){
18         swap(ans[i].first, ans[i].second);
19     }
20 }
21 sort(ans.begin(), ans.end());
22 cout << v[0].first + v.back().first << endl;
23 for(int i = 0; i < n; i++){
24     cout << ans[i].first << ' ' << ans[i].second << endl;
25 }
26 }
```

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 вытаскивание 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 $2 \leq a \leq 10^{18}$, $1 \leq b$, $c \leq 10^{18}$.

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 каждого 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++

```

1  #include<bits/stdc++.h>
2  #define int long long
3  using namespace std;
4  int f1(int M, int a){
5      int res = 0, z = 0;
6      while(1){
7          if(M == 0 && z < a){
8              return res;
9          }
10         res += M;
11         M = M + z;
12         z = M % a;
13         M = M / a;
14     }
15 }
```

```

16 int f2(int M, int b){
17     return M + M / b;
18 }
19 signed main(){
20     int a, b, c;
21     cin >> a >> b >> c;
22     int L = 0, R = 1;
23     while(f1(R, a) <= c){
24         R *= 2;
25     }
26     while(R - L > 1){
27         int M = (R + L) / 2;
28         if(f1(M, a) < c){
29             L = M;
30         }
31         else{
32             R = M;
33         }
34     }
35     int z = R;
36     L = 0, R = 1;
37     while(f2(R, b) <= z){
38         R *= 2;
39     }
40     while(R - L > 1){
41         int M = (R + L) / 2;
42         if(f2(M, b) < z){
43             L = M;
44         }
45         else{
46             R = M;
47         }
48     }
49     cout << R << endl;
50 }

```

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, заработка 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 $2 \leq n, m \leq 2 \cdot 10^5$.

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. $1 \leq A, B \leq n$, $1 \leq d \leq 10^9$. 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 , $1 \leq S \leq n$.

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++

```
1 #include<bits/stdc++.h>
2 #define int long long
3 using namespace std;
4 int n, m;
5 vector<vector<pair<int, int> > > G;
6 vector<int> order, used;
7 void dfs(int a){
8     used[a] = 1;
9     for(auto to : G[a]){
10         if(!used[to.first]){
11             dfs(to.first);
12         }
13     }
14     order.push_back(a);
15 }
16 signed main(){
17     cin >> n >> m;
18     G.resize(n + 1);
19     used.resize(n + 1, 0);
20     for(int i = 0; i < m; i++){
21         int a, b, d;
22         cin >> a >> b >> d;
23         G[a].push_back({b, d});
24     }
}
```

```

25     int s;
26     cin >> s;
27     dfs(s);
28     reverse(order.begin(), order.end());
29     vector<int> dp(n + 1, -1);
30     dp[s] = 0;
31     for(auto el : order){
32         for(auto to : G[el]){
33             dp[to.first] = max(dp[to.first], dp[el] + to.second);
34         }
35     }
36     for(int i = 1; i <= n; i++){
37         cout << dp[i] << ' ';
38     }
39 }

```

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 $n \times m$.

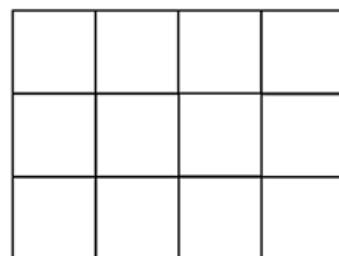
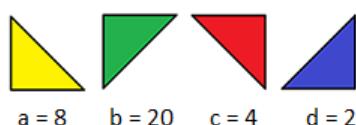


Рис. 2.2.1

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.

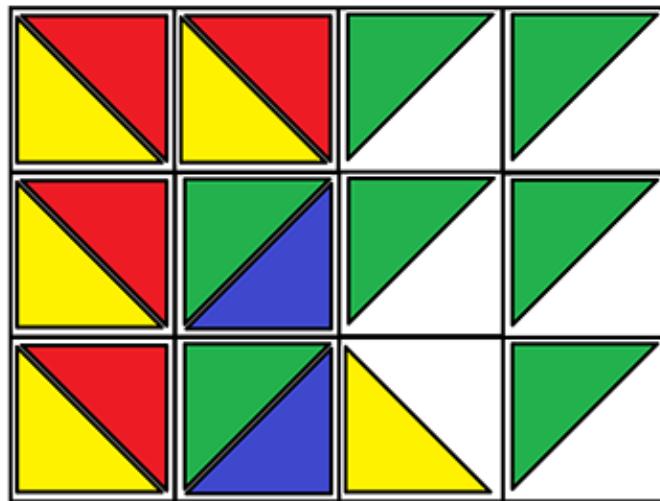


Рис. 2.2.2

Решение

Ниже представлено решение на языке C++.

C++

```

1 #include<bits/stdc++.h>
2 #define int long long
3 using namespace std;
4 signed main(){
5     int a, b, c, d, n, m;
6     cin >> a >> b >> c >> d >> n >> m;
7     if(a > c){
8         swap(a, c);
9     }
10    if(b > d){
11        swap(b, d);
12    }
13    int f = a + b;
14    int k = n * m;
15    if(k <= f){
16        cout << k * 2;
17        return 0;
18    }
19    k -= f;
20    c -= a;
21    d -= b;
22    cout << f * 2 + min(k, c + d) << endl;
23 }
```

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 $1 \leq n \cdot k \leq 10^{18}$.

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++

```

1 #include<bits/stdc++.h>
2 #define int long long
3 using namespace std;
4 signed main(){
5     int n, k;
6     cin >> n >> k;
7     int x = (k * n) / (k * k + 2 * k + 1);
8     cout << x - k << ' ' << x + k << ' ' << x / k << ' ' << x * k << endl;
9 }
```

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, $2 \leq n, m \leq 25$. Во второй строке через пробел находятся координаты (x, y) обязательной для посещения клетки, где $1 \leq x \leq n$, $1 \leq y \leq m$. 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.

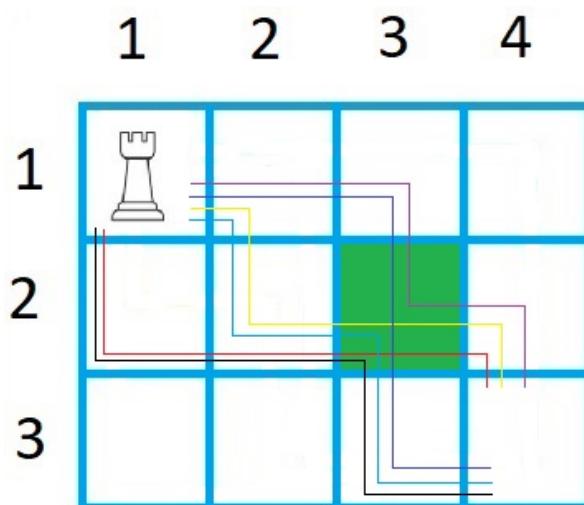


Рис. 2.2.3

Решение

Ниже представлено решение на языке C++.

C++

```

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

```

```

9         bc[i][j] += bc[i - 1][j];
10        if(j - 1 >= 0){
11            bc[i][j] += bc[i - 1][j - 1];
12        }
13    }
14
15    int n, m, x, y;
16    cin >> n >> m >> x >> y;
17    int d1 = bc[x - 1 + y - 1][x - 1];
18    int d2 = bc[n - x + m - y][n - x];
19    int ans = d1 * d2;
20    cout << ans << endl;
21 }

```

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 футбольку, 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 левые 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++

```

1 #include<bits/stdc++.h>
2 #define int long long
3 using namespace std;
4 signed main(){
5     string s;
6     cin >> s;
7     string mx = s, mn = s;
8
9     for(int i = 0; i < 5; i++){
10         for(int j = 0; j < 10; j++){
11             mx = max(mx, s);
12             mn = min(s, mn);
13             if(j < 9){
14                 s = s.substr(1) + s[0];
15             }
16         }
17         for(int j = 0; j < 5; j++){
18             swap(s[2 * j], s[2 * j + 1]);
19         }
20     }
21     stringstream ssmn;
22     ssmn << mn;
23     int imn;
24     ssmn >> imn;
25     stringstream ssmx;
```

```

26     ssmx << mx;
27     int imx;
28     ssmx >> imx;
29     cout << imx - imn << endl;
30 }
```

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 $1, \dots, n$.

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 $1 \leq n \leq 2 \cdot 10^5$.

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++

```

1  #include<bits/stdc++.h>
2  #define int long long
3  using namespace std;
4  const int LG = 19;
5  int N = (1 << LG);
6  vector<int> tr(2 * N, 0);
7  void upd(int pos, int x){
8      pos += N;
9      tr[pos] = x;
10     pos /= 2;
11     while(pos){
12         tr[pos] = {tr[2 * pos] + tr[2 * pos + 1]};
13         pos /= 2;
14     }
15 }
16 int get(int l, int r){
17     l += N;
18     r += N;
19     int res = 0;
20     while(l <= r){
21         if(l % 2 == 1){
22             res += tr[l];
23         }
24         if(r % 2 == 0){
25             res += tr[r];
26         }
27         l = (l + 1) / 2;
28         r = (r - 1) / 2;
29     }
30     return res;
31 }
32 signed main(){
33     int n, a;
34     cin >> n;
35     int ans = 0;
36     for(int i = 0; i < n; i++){
37         cin >> a;
38         int sl = get(0, a - 1);
39         int sr = get(a + 1, N - 1);
40         ans += min(sl, sr);
41         upd(a, a);
42     }
43     cout << ans << endl;
44 }
```

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++

```

1 #include<bits/stdc++.h>
2 #define int long long
3 using namespace std;
4 signed main(){
5     int A, C, D;
6     cin >> A >> C >> D;
7     cout.precision(1);
8     cout << fixed << (A + C) / 2.0 << endl;
9 }
```

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

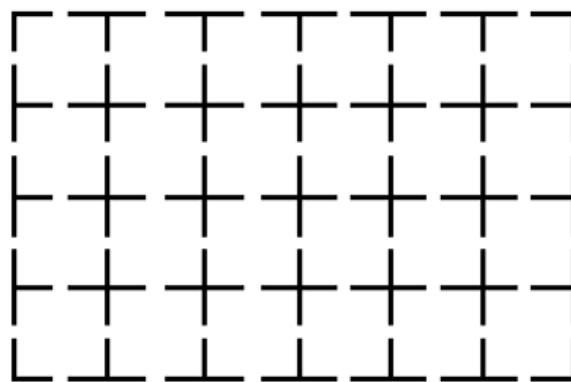


Рис. 2.2.4

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 $1 \leq n, m \leq 10^9$.

Формат выходных данных

Требуется по числам a и b найти размеры исходной решетки n и m . Будем всегда считать, что $n \leq m$, 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++

```

1 #include<bits/stdc++.h>
2 #define int long long
3 using namespace std;
4 signed main(){
5     int a, b;
6     cin >> a >> b;
7     int L = 0, R = a / 4 + 1;
8     while(R - L > 1){
9         int M = (R + L) / 2;
10        int D = a / 2 - M;
11        if(M * D <= b){
12            L = M;
13        }
14        else{
15            R = M;
16        }
17    }
18    cout << L + 1 << ' ' << a / 2 - L + 1 << endl;
19 }
```

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 учи, 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 $2 \leq n, m \leq 1000$.

Во второй строке находятся два числа x и y через пробел — координаты свечки, где $1 \leq x \leq n - 1$, $1 \leq y \leq m - 1$, 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.

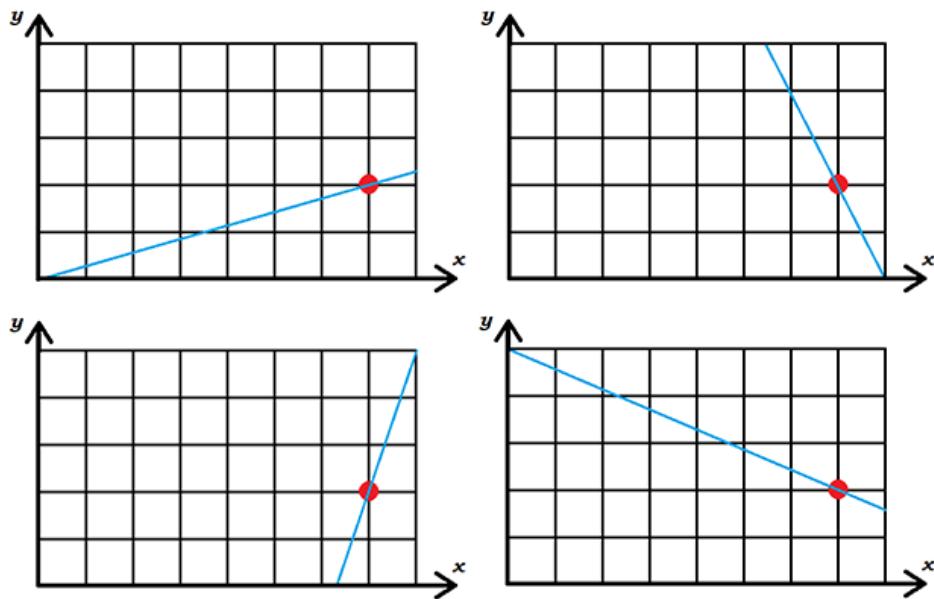


Рис. 2.2.5

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++

```

1  #include<bits/stdc++.h>
2  #define int long long
3  using namespace std;
4  const int INF = 1e18;
5  double katy(double x, double y, double n){
6      return n * y / x;
7  }
8  double n, m, x, y;
9  double ans = INF;
10 double k1, k2;
11 void upd(){
12     if(k1 < m){
13         double st = k1 * n / 2;
14         ans = min(ans, n * m - 2 * st);
15     }
16     else{
17         double st = k2 * m / 2;
18         ans = min(ans, n * m - 2 * st);
19     }
20 }
21 signed main(){
22     cin >> n >> m >> x >> y;
23     k1 = katy(x, y, n);
24     k2 = katy(y, x, m);
25     upd();
26     k1 = katy(n - x, y, n);
27     k2 = katy(y, n - x, m);
28     upd();
29     k1 = katy(x, m - y, n);
30     k2 = katy(m - y, x, m);
31     upd();
32     k1 = katy(n - x, m - y, n);
33     k2 = katy(m - y, n - x, m);
34     upd();
35     cout.precision(3);
36     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 $2 \leq n \leq 2 \cdot 10^5$. Во второй строке через пробел заданы n целых чисел a_i через пробел — длины полосок, где $1 \leq a_i \leq 10^9$.

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.

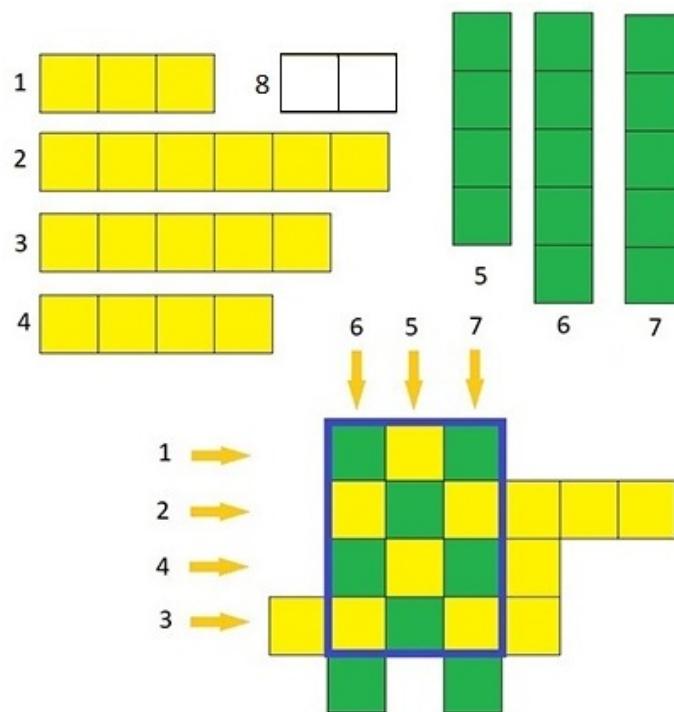


Рис. 2.2.6

Решение

Ниже представлено решение на языке C++.

C++

```
1 #include<bits/stdc++.h>
2 #define int long long
3 using namespace std;
4 signed main(){
5     int n;
6     cin >> n;
7     deque<int> v(n);
8     for(int i = 0; i < n; i++){
9         cin >> v[i];
10    }
11    sort(v.begin(), v.end());
12    int ans = 0;
13    int cnth = 0, minh;
14    while(1){
15        if(v.size() == 0){
16            break;
17        }
18        cnth++;
19        minh = v.back();
20        v.pop_back();
21        while(v.size() > 0 && v[0] < cnth){
22            v.pop_front();
23        }
24        ans = max(ans, cnth * min(minh, (int)v.size()));
25    }
26    cout << ans << endl;
27 }
```

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 $20 \leq n \leq 100$, 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, $20 \leq n \leq 100$.

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++

```

1 #include<bits/stdc++.h>
2 #define int long long
3 using namespace std;
4 string frs;
5 int n;
6 vector<string> dict;
7 vector<int> msk(26, 0);
8 int cnt = 0;
9 vector<vector<int>> amsk;
10 vector<string> ans;
11 bool bigok = 0;
12 void p(int pos){
13     if(!bigok){
14         if(cnt == 0){
15             sort(ans.begin(), ans.end());
16             bigok = 1;
17             return;
18         }
19         for(int i = pos; i < n; i++){
20             string ts = dict[i];
21             bool ok = 1;
22             for(int j = 0; j < 26; j++){
23                 if(amsk[i][j] > msk[j]){
24                     ok = 0;
25                 }
26             }
27             if(ok){
28                 ans.push_back(ts);
29                 for(int j = 0; j < 26; j++){
30                     msk[j] -= amsk[i][j];
31                     cnt -= amsk[i][j];
32                 }
33                 p(i + 1);
34                 if(!bigok){
35                     for(int j = 0; j < 26; j++){
36                         msk[j] += amsk[i][j];
37                         cnt += amsk[i][j];
38                     }
39                     ans.pop_back();
40                 }
41             }
42         }
43     }
44 }
45 signed main(){
46     cin >> frs;
47     cin >> n;
48     amsk.resize(n, vector<int>(26, 0));
49
50     string ts;
51     for(int i = 0; i < n; i++){
52         cin >> ts;
53         dict.push_back(ts);
54     }
55     for(int i = 0; i < n; i++){
56         for(auto el : dict[i]){
57             amsk[i][el - 'a']++;
58         }
59     }
}

```

```

60     for(auto el : frs){
61         msk[el - 'a']++;
62         cnt++;
63     }
64     p(0);
65     for(auto el : ans){
66         cout << el << endl;
67     }
68 }
```

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 $1 \leq a, b, c, d \leq 10^9$.

Формат выходных данных

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

Примеры

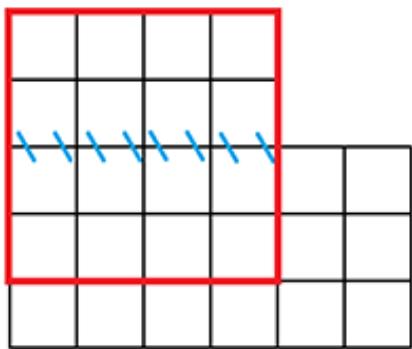
Пример №1

| Стандартный ввод |
|-------------------|
| 2 4 |
| 3 6 |
| Стандартный вывод |
| 4 |

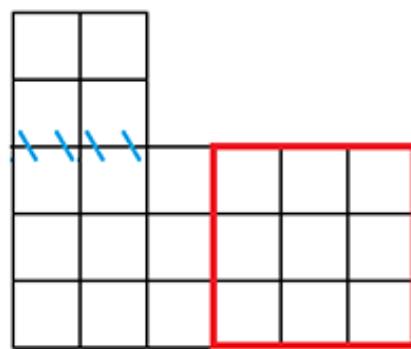
Пример №2

| Стандартный ввод |
|-------------------|
| 2 2 |
| 3 6 |
| Стандартный вывод |
| 3 |

Примечания



$$2 \times 4 + 3 \times 6$$



$$2 \times 2 + 3 \times 6$$

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++

```

1 #include<bits/stdc++.h>
2 #define int long long
3 using namespace std;
4 signed main(){
5     int a, b, c, d;
6     cin >> a >> b >> c >> d;
7     int ans = max(min(a, b), min(c, d));
8     int p1 = min(a + c, min(b, d));
9     int p2 = min(a + d, min(b, c));
10    int p3 = min(b + c, min(a, d));
11    int p4 = min(b + d, min(a, c));
12    ans = max({ans, p1, p2, p3, p4});
13    cout << ans << endl;
14 }

```

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 А, С, Г, Т.

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 префиксам длины i .

Например, пусть $D_1 = \texttt{AACGGTAGGT}$, а $D_2 = \texttt{TCCCGGAACA}$. Выберем $i = 4$ и поменяем местами префиксы длины 4. Получим две новые ДНК, одна из которых будет иметь вид $\texttt{AACGGGAACA}$, а вторая — $\texttt{TCCCGTAGGT}$. Для наглядности были выделены части первой из них.

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

Теперь можно рассмотреть популяцию из нескольких ДНК. Выберем из них две, произведем их скрещивание по префиксам какого-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 t ДНК, каждая имеет одну и ту же длину n . После некоторого количества парных 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 $t - 1$ 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 $10 \leq n \leq 100$, $2 \leq m \leq 100$.

В следующих m строках содержится описание исходной популяции ДНК, каждая задается строкой длины n , состоящей из символов А, С, Г и Т.

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

```
AATTCTTGAA
```

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++

```

1  #include<bits/stdc++.h>
2  #define int long long
3  using namespace std;
4  signed main(){
5      int n, m;
6      cin >> n >> m;
7      vector<string> v1(m);
8      for(int i = 0; i < m; i++){
9          cin >> v1[i];
10     }
11     string d;
12     cin >> d;
13     vector<string> v2(m - 1);
14     for(int i = 0; i < m - 1; i++){
15         cin >> v2[i];
16     }
17     for(int j = 0; j < n; j++){
18         int ss = 0;
19         for(int i = 0; i < m; i++){
20             ss ^= (int)(v1[i][j]);
21         }
22         for(int i = 0; i < m - 1; i++){
23             ss ^= (int)(v2[i][j]);
24         }
25         cout << (char)(ss);
26     }
27     cout << endl;
28 }
```

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 временем 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 $1 \leq A, B, C, D \leq 1000$, $0 \leq E \leq 1000$.

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++

```

1 #include<bits/stdc++.h>
2 #define int long long
3 using namespace std;
4 const int INF = 1e18;
5 vector<int> v(4);
6 int e, ans = INF;
7 void p(vector<int> &vl, vector<int> &vr, int tv){
8     if(vl.size() == 2){
9         ans = min(ans, tv + *max_element(vl.begin(), vl.end()));
10        return;
11    }
12    for(int i = 0; i < vl.size() - 1; i++){
13        for(int j = i + 1; j < vl.size(); j++){
14            vector<int> vl1;
15            for(int k = 0; k < vl.size(); k++){
16                if(k != i && k != j){
17                    vl1.push_back(vl[k]);
18                }
19            }
20            vector<int> vr1 = vr;

```

```

21         vr1.push_back(vl[i] + e);
22         vr1.push_back(vl[j] + e);
23         int tmp = max(vl[i], vl[j]);
24         sort(vr1.rbegin(), vr1.rend());
25         vl1.push_back(vr1.back() + e);
26         vr1.pop_back();
27         p(vl1, vr1, tv + tmp + vl1.back() - e);
28     }
29 }
30
31 signed main(){
32     for(int i = 0; i < 4; i++){
33         cin >> v[i];
34     }
35     sort(v.begin(), v.end());
36     cin >> e;
37     vector<int> vl = v, vr;
38     p(vl, vr, 0);
39     cout << ans;
40 }

```

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 $2 \leq a \leq 10^6$, $0 \leq b \leq 10^6$.

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++

```

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

```

```

7  int binpow (int a, int n) {
8      int res = 1;
9      while (n > 0) {
10         if (n % 2 == 1)
11             (res *= a) %= MOD;
12         (a *= a) %= MOD;
13         n /= 2;
14     }
15     return res;
16 }
17
18 int bc(int n, int k){
19     int res = f[n];
20     int p1 = binpow(f[k], MOD - 2);
21     int p2 = binpow(f[n - k], MOD - 2);
22     (res *= p1) %= MOD;
23     (res *= p2) %= MOD;
24     return res;
25 }
26 signed main(){
27     for(int i = 1; i <= 2e6; i++){
28         f[i] = (f[i - 1] * i) % MOD;
29     }
30     int a, b;
31     int ans = 0;
32     cin >> a >> b;
33     a--;
34     for(int i = 0; i < a + 1; i++){
35         if(2 * i <= b){
36             int d = bc(a, i);
37             if(b - 2 * i <= a - i){
38                 (d *= bc(a - i, b - 2 * i)) %= MOD;
39                 (ans += d) %= MOD;
40             }
41         }
42     }
43     cout << ans << endl;
44 }
```

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 $1 \leq d \leq 200$, $2 \leq n \leq 2000$.

Во второй строке находятся n целых чисел v_i через пробел — скорости каждого джентльмена, где $1 \leq v_i \leq 2000$. Гарантируется, что все скорости попарно различны. Скорости даны в порядке возрастания, то есть $v_1 < v_2 < \dots < v_n$.

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

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.

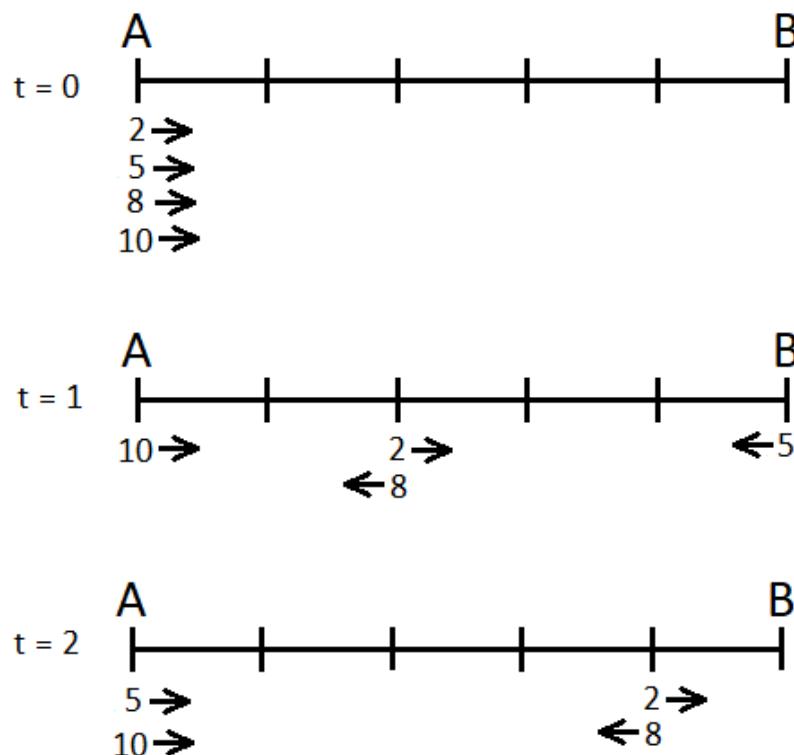


Рис. 2.2.8

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 встречаются 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++

```

1  #include<bits/stdc++.h>
2  #define int long long
3  using namespace std;
4  const double EPS = 1e-7;
5  double x(double M, int V, int d){
6      double dst = V * M;
7      int cnt = floor((dst + EPS) / d);
8      double pin = dst - cnt * d;
9      if(cnt % 2 == 0){
10         return pin;
11     }
12     else{
13         return d - pin;
14     }
15 }
16 int F(double M, vector<int> &v, int d){
17     int res = 0;
18     for(int i = 0; i < v.size(); i++){
19         double dst = v[i] * M;

```

```

20     int cnt = floor((dst + EPS) / d);
21     res += cnt * i;
22     double tx = x(M, v[i], d);
23     for(int j = 0; j < i; j++){
24         double txj = x(M, v[j], d);
25         if(cnt % 2 == 0){
26             res += txj <= tx + EPS;
27         }
28         else{
29             res += txj >= tx - EPS;
30         }
31     }
32 }
33 return res;
34 }
35 signed main(){
36     int d, n;
37     cin >> d >> n;
38     vector<int> v(n);
39     for(int i = 0; i < n; i++){
40         cin >> v[i];
41     }
42     int k;
43     cin >> k;
44     double L = 0, R = 1;
45     while(F(R, v, d) <= k){
46         R *= 2;
47     }
48     R *= 2;
49     while(R - L > 1e-4){
50         double M = (R + L) / 2.0;
51         if(F(M, v, d) < k){
52             L = M;
53         }
54         else{
55             R = M;
56         }
57     }
58     cout.precision(10);
59     cout << fixed << L << endl;
60 }
```

2.3. Предметный тур. Физика

2.3.1. Первая волна. Задачи 8–9 класса

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

Задача 2.3.1.1. Калориметр (10 баллов)

Условие

Внутренний стакан калориметра представляет собой цилиндр с радиусом $R = 8\text{ см}$ 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 $\rho = 25\text{ кг}/\text{м}^3$ необходима, чтобы полностью заполнить пространство между стаканами?

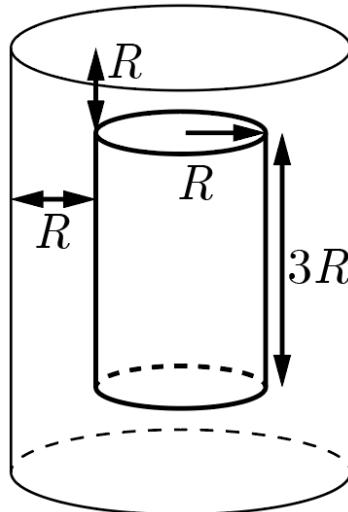


Рис. 2.3.1

Решение

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

$$m = \rho V. \quad (2.3.1)$$

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,

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

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

$$V_2 = 3\pi R^3. \quad (2.3.3)$$

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

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

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

$$P = U^2/R. \quad (2.3.5)$$

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

$$R = \frac{\rho l}{S}. \quad (2.3.6)$$

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 $h = 1,4$ м и массой $m = 160$ 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 $F = 4$ кН. Ускорение свободного падения $g \approx 9,8$ м/с².

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

$$A = mgh + \frac{Fh}{2} \approx 5 \text{ кДж.} \quad (2.3.7)$$

Погрешность 0,1 кДж.

Ответ: $(5,0 \pm 0,1)$ 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 $v_1 = 15$ м/с, а через время $\tau = 40$ с быстро переключается на движение с постоянной скоростью $v_2 = 20$ 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 $\Delta v = 1$ 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 :

$$v = \frac{S}{t}. \quad (2.3.8)$$

Для первого дрона это уравнение принимает вид

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

а для второго, соответственно,

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

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

$$v_1\tau + v_2(t - \tau) - v_2\tau - v_1(t - \tau) = \Delta vt. \quad (2.3.11)$$

Перегруппировав слагаемые, получим

$$v_2t - 2v_2\tau - v_1t + 2v_1\tau = \Delta vt, \quad (2.3.12)$$

откуда окончательно

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

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.

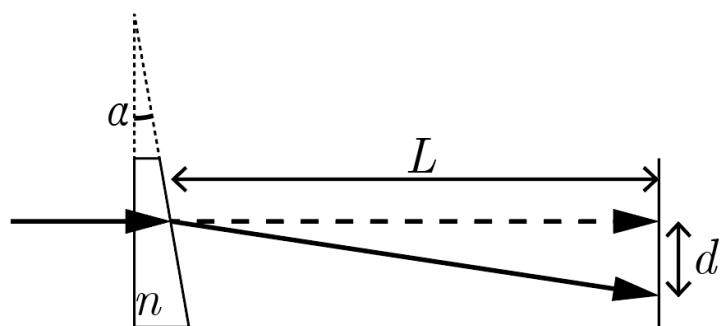


Рис. 2.3.2

Определите показатель преломления стекла, из которого изготовлена призма, если расстояние от передней грани призмы до экрана $L = 3 \text{ м}$, смещение пятна при установке призмы $d = 12 \text{ см}$, а угол между передней и задней поверхностями призмы $\alpha = 3^\circ$. 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

$$n = \frac{\sin \beta}{\sin \alpha}, \quad (2.3.14)$$

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

$$n \approx \frac{\beta}{\alpha}. \quad (2.3.15)$$

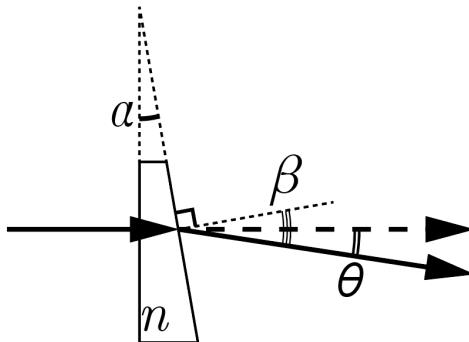


Рис. 2.3.3

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

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

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

$$p_0 = \frac{mL}{t_0}, \quad (2.3.17)$$

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

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

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

$$t = \frac{\cancel{L}p_0t_0}{\cancel{L}(p_0 - \Delta p)} = 100 \text{ с.} \quad (2.3.19)$$

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

$$F = ma_0, \quad (2.3.20)$$

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

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

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

$$t = \sqrt{\frac{2l}{a - a_0}} = 5 \text{ с.} \quad (2.3.22)$$

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 $R = 5$ кОм. Таня же решила, что это конденсаторы и, соединив их, получила элемент с эквивалентной емкостью $C = 2$ мкФ. Определите, чему равнялось число y .

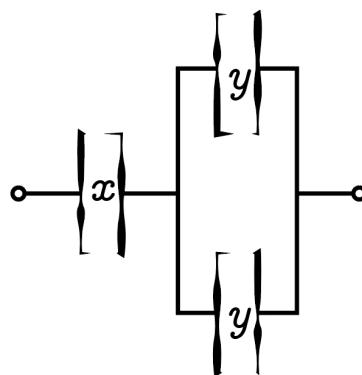


Рис. 2.3.4

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

$$R = x + \frac{y}{2}. \quad (2.3.23)$$

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

$$C = \frac{2xy}{x + 2y}. \quad (2.3.24)$$

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

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

Домножив полученное уравнение на знаменатель дроби и раскрыв скобки, получим

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

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

$$10 + 3y = 10y - y^2. \quad (2.3.27)$$

Это квадратное уравнение, которое легко привести к канонической форме

$$y^2 - 7y + 10 = 0 \quad (2.3.28)$$

и решить с помощью дискриминанта

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

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

$$\eta = \frac{T_2 - T_1}{T_2}, \quad (2.3.30)$$

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

$$\frac{T_2 - T_1}{T_2} = \eta = \frac{A}{Q_2}. \quad (2.3.31)$$

Отсюда Q_2 может быть выражено как

$$Q_2 = \frac{AT_2}{T_2 - T_1}. \quad (2.3.32)$$

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

$$Q_1 = Q_2 - A = \frac{AT_1}{T_2 - T_1}. \quad (2.3.33)$$

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

$$\lambda m = \frac{AT_1}{\theta - T_1}, \quad (2.3.34)$$

откуда окончательно выражим ответ

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

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 $I_1 = 10 \text{ A}$ он нагревается до температуры $t_1 = 70^\circ\text{C}$, после чего перестает увеличивать свою температуру, а при пропускании через него тока $I_2 = 20 \text{ A}$ — до температуры $t_2 = 205^\circ\text{C}$.

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 коэффициент этой пропорциональности κ

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

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

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

где R — сопротивление катушки.

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

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

Тривиальными алгебраическими преобразованиями выразим θ

$$(\theta - t_1)I_2^2 = (\theta - t_2)I_1^2 \Rightarrow \theta = \frac{t_1 I_2^2 - t_2 I_1^2}{I_2^2 - I_1^2} = 25^\circ\text{C}. \quad (2.3.39)$$

Погрешность: $0,1^\circ\text{C}$.

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.Конкурс: <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 $c_{уд}$ 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 | $\rho, \text{ г/см}^3$ | $c_{уд}, \text{ кДж/(кг} \cdot {^\circ}\text{C)}$ |
|---|--------------|------------------------|---|
| 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 |

Ведите 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

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

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

$$m = \rho V, \quad (2.3.41)$$

откуда

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

Поскольку величины V, θ, t_0 независимы от выбранного вещества, задача сводится к расположению в порядке возрастания произведений $c_{уд}\rho$. 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.

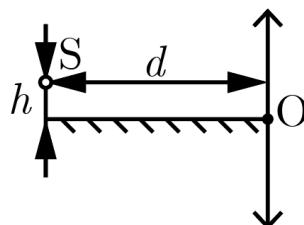


Рис. 2.3.5

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).

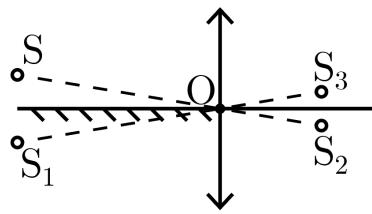


Рис. 2.3.6

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

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

Искомое расстояние l между изображениями S_2 и S_3 , благодаря подобию треугольников $\triangle OSS_1$ и $\triangle OS_2S_3$, 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

$$\frac{l}{2h} = \frac{f}{d} = \frac{F}{d - F}. \quad (2.3.44)$$

Отсюда окончательно находим

$$l = 2h \frac{F}{d - F} = 2 \text{ см.} \quad (2.3.45)$$

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.

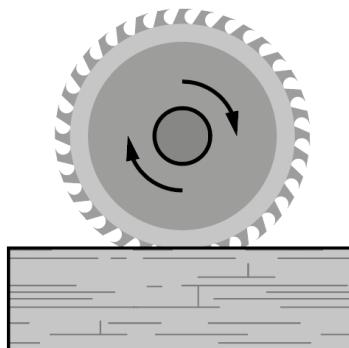


Рис. 2.3.7

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

$$A = FS = Fvt, \quad (2.3.46)$$

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

$$v = \frac{\pi D}{T} = \pi D\nu, \quad (2.3.47)$$

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

$$Q = \pi F D \nu t \Rightarrow F = \frac{Q}{\pi D \nu t} \approx 37 \text{ Н.} \quad (2.3.48)$$

Погрешность 1 Н.

Ответ: $F = (37 \pm 1)$ 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

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

Сократив v_0 и перегруппировав слагаемые, получим

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

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

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

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

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

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

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

Отсюда окончательно

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

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.

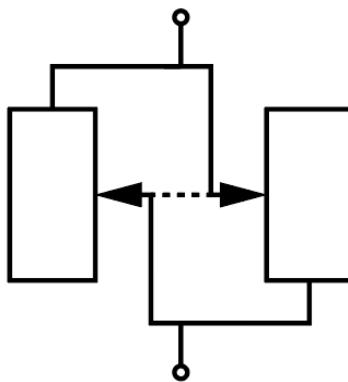


Рис. 2.3.8

Решение

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

$$R = \frac{R_1 R_2}{R_1 + R_2}. \quad (2.3.55)$$

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

$$R_2 = R_0 - R_1. \quad (2.3.56)$$

Подставляя этот результат в (2.3.55), получим

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

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 «поворнута» 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 $R_1 = R_2 = \frac{R_0}{2}$) сопротивление батареи равно $\frac{R_0}{4}$.

Таким образом,

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

Погрешность 0,01 кОм.

Ответ: $l = (0,50 \pm 0,01) \text{ кОм.}$

2.3.4. Вторая волна. Задачи 10–11 класса

Задачи второй волны предметного тура 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

$$q_1 = 2,5\tau I_0, \quad (2.3.59)$$

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

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

Складывая эти заряды, получим окончательно

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

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.

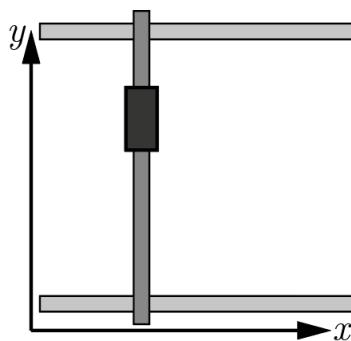


Рис. 2.3.9

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

$$v = \frac{2\pi R}{T}. \quad (2.3.62)$$

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

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

Погрешность 0,1 мм/с².

Ответ: $(2,0 \pm 0,1)$ мм/с².

Задача 2.3.4.3. Плита (20 баллов)

Условие

Квадратная плита ABCD со стороной $a = 40$ 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 С 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?

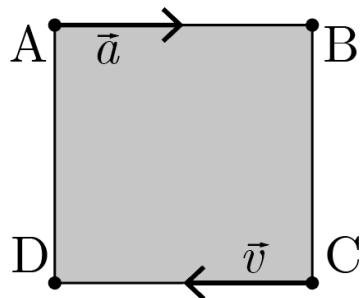


Рис. 2.3.10

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 С 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

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

Погрешность 0,5 см.

Ответ: $(28,3 \pm 0,5)$ 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.

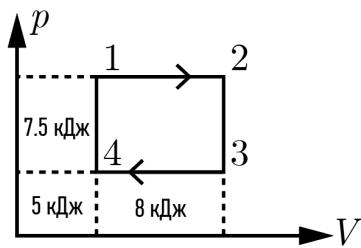


Рис. 2.3.11

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:

$$\eta = \frac{A}{Q}. \quad (2.3.65)$$

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

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

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

$$A_{412} = 12 + 8 = 20 \text{ кДж}, \quad (2.3.67)$$

а внутренняя энергия может быть удобно вычислена по формуле

$$U = \frac{3}{2}PV, \quad (2.3.68)$$

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

$$U_4 = \frac{3}{2}5 = 7,5 \text{ кДж}, \quad (2.3.69)$$

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

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

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

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

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?

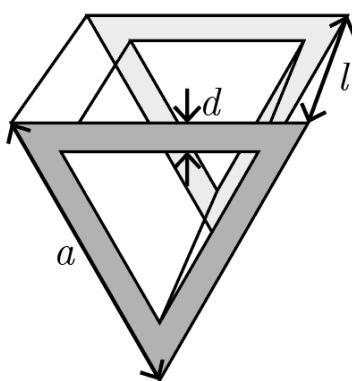


Рис. 2.3.12

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.

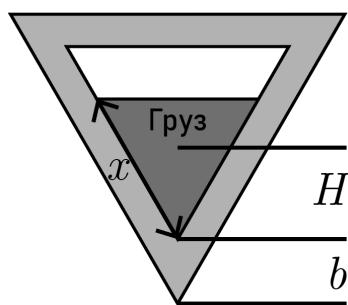


Рис. 2.3.13

Отсчитывая от киля (нижней вершины треугольника) высоту $y_{\text{в}}$, на которой

находится центр тяжести вытесненной воды, легко выразить как

$$y_{\text{в}} = \frac{\sqrt{3}}{3}a, \quad (2.3.72)$$

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

Высота y_{Γ} центра тяжести груза может быть найдена как

$$y_{\Gamma} = b + H, \quad (2.3.73)$$

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

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

где x — сторона треугольника, являющегося поперечным сечением груза.

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

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

откуда

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

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

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

Погрешность 10 м³.

Ответ: $(696 \pm 10) \text{ м}^3$.

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.Конкурс: <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 $\rho = 900 \text{ кг}/\text{м}^3$. 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 $g = 9,8 \text{ м}/\text{с}^2$.

Решение

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

$$p = \rho gh. \quad (2.3.78)$$

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

$$F = pS = \rho ghS. \quad (2.3.79)$$

Искомая расчетная 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

$$f = \frac{3F}{16} = \frac{3\rho ghS}{16} \approx 9,9 \text{ кН.} \quad (2.3.80)$$

Погрешность 0,1 кН.

Ответ: $(9,9 \pm 0,1)$ кН.

Задача 2.3.5.2. Реостат (15 баллов)

Условие

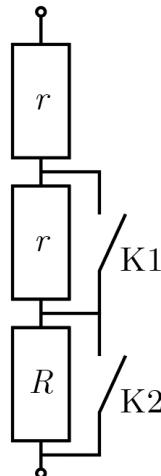


Рис. 2.3.14

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 $R_1 = 2 \text{ кОм}$, а напротив, при замыкании ключа K_2 и размыкании ключа K_1 — сопротивление $R_2 = 3 \text{ кОм}$. 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

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

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

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

Погрешность 0,01 кОм.

Ответ: $(3,50 \pm 0,1)$ 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 $c = 155 \text{ Дж}/(\text{кг}\cdot\text{°C})$ и средней плотностью $\rho = 10^4 \text{ кг}/\text{м}^3$, если, двигаясь в трубе диаметром $d = 10 \text{ см}$ со скоростью $v = 20 \text{ м}/\text{с}$, 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 из реактора уходит объем

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

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

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

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.

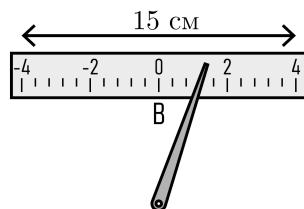


Рис. 2.3.15

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).

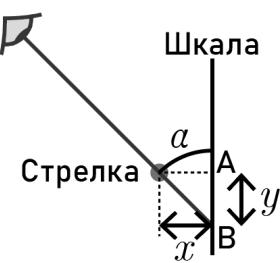


Рис. 2.3.16

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

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

Отсюда окончательно

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

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 А 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

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

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

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

откуда легко найти

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

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

$$\frac{v_3}{v_1} = \frac{S_3}{S_1} = \frac{\sqrt{a^2 + b^2} + a}{b} = \frac{\sqrt{\frac{25a^2}{9} + a}}{4a} = \frac{\frac{5a}{3} + a}{4a} = \frac{\frac{8a}{3}}{4a} = \frac{2}{3} = 2. \quad (2.3.90)$$

Погрешность 0,01.

Ответ: $2,00 \pm 0,01$.

2.3.6. Третья волна. Задачи 10–11 класса

Задачи третьей волны предметного тура по 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 $T_0 = -273^\circ\text{C}$. 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, $t_x = 0^\circ\text{C}$. Аналогично, при атмосферном давлении температура кипящей воды может быть равна только $t_h = 100^\circ\text{C}$. КПД идеальной тепловой машины (машины Карно) с известными абсолютными термодинамическими температурами T_h и T_x холодильника задается выражением

$$\eta_0 = \frac{T_h - T_x}{T_h}. \quad (2.3.91)$$

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

$$\eta = \frac{\eta_0}{\alpha} = \frac{t_h - t_x}{\alpha(t_h - T_0)} \approx 16,8\%. \quad (2.3.92)$$

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 $v = 12 \text{ м/с}$ относительно поверхности воды. Наблюдательный дрон запрограммирован летать на постоянной высоте по траектории, в системе отсчета корабля представляющей собой окружность с радиусом $R = 2 \text{ км}$ 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 $v_{\text{max}} = |\vec{v}_{\text{дк}}| + |\vec{v}_{\text{кв}}|$ и $v_{\text{min}} = ||\vec{v}_{\text{дк}}| - |\vec{v}_{\text{кв}}||$ 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

$$v_{\text{дк}} = \frac{2\pi R}{T}. \quad (2.3.93)$$

Отсюда получим окончательно

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

Погрешность 0,02.

Ответ: $3,68 \pm 0,02$.

Задача 2.3.6.3. Конденсатор (20 баллов)

Условие

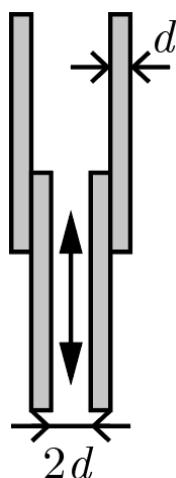


Рис. 2.3.17

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

$$C_1 = \frac{S\epsilon_0}{2d}, \quad (2.3.95)$$

где ϵ_0 — диэлектрическая постоянная.

Аналогично, емкость полностью выдвинутой внешней пары равна

$$C_2 = \frac{S\epsilon_0}{4d}, \quad (2.3.96)$$

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

Тогда для искомого отношения максимальной и минимальной емкостей получим:

$$\frac{C_{max}}{C_{min}} = \frac{C_1 + C_2}{C_1} = \frac{S\epsilon_0/(4d) + S\epsilon_0/(2d)}{S\epsilon_0/(2d)} = \frac{1/4 + 1/2}{1/2} = 1,5. \quad (2.3.97)$$

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 $\Delta t = 70^\circ\text{C}$ выше, чем температура окружающего шара воздуха. Аэростат имеет объем $V = 645 \text{ м}^3$, а общая масса его оболочки, корзины и полезной нагрузки $M = 150 \text{ кг}$. Определите, при какой максимальной температуре окружающей среды аэростат сможет взлететь?

Абсолютный ноль температур $T_0 = -273^{\circ}\text{C}$, 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

$$p_0 V = \frac{m}{\mu} RT, \quad (2.3.98)$$

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

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

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

где T_0 — искомая температура окружающего воздуха.

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

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

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

$$\mu p_0 V (T_0 + \Delta t) = M RT_0 (T_0 + \Delta t) + \mu p_0 V T_0. \quad (2.3.101)$$

В канонической форме:

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

Остается найти (методом дискриминанта) единственный положительный корень этого уравнения

$$T_0 = -\frac{\Delta t}{2} + \frac{1}{2} \sqrt{\Delta t^2 + \frac{4\mu p_0 V}{MR} \Delta t} \approx 291 \text{ K} \approx 18^{\circ}\text{C}. \quad (2.3.103)$$

Погрешность $0,1^{\circ}\text{C}$.

Ответ: $(18,0 \pm 0,1)^{\circ}\text{C}$.

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

$$F = G \frac{mM}{R^2}, \quad (2.3.104)$$

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

$$m \frac{v^2}{R} = G \frac{mM}{R^2}. \quad (2.3.105)$$

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.Конкурс: <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 $\lambda = 28,7 \text{ кДж/кг}$, ее теплоемкость в твердой форме $c_1 = 0,92 \text{ кДж/(кг}^{\circ}\text{C)}$, а в жидкой — $c_2 = 1,5 \text{ кДж/(кг}^{\circ}\text{C)}$, температура ее плавления $\theta = 800 \text{ }^{\circ}\text{C}$. Определите, какую массу соли необходимо взять, чтобы при ее нагреве от $t_0 = 20 \text{ }^{\circ}\text{C}$ до $t_1 = 1200 \text{ }^{\circ}\text{C}$ запасти $Q = 1 \text{ МДж}$ тепла.

Решение

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

$$Q_1 = c_1 m(\theta - t_0), \quad (2.3.106)$$

где m — масса нагреваемой соли.

Количество теплоты, уходящее непосредственно на плавление

$$Q_2 = \lambda m. \quad (2.3.107)$$

Наконец, количество теплоты, уходящее на нагрев расплава соли,

$$Q_3 = c_2 m(t_1 - \theta). \quad (2.3.108)$$

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

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

откуда окончательно

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

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 $P = 5 \text{ мВт}$. Какое количество световой энергии поглотит второе зеркало за $\tau = 10 \text{ с}$?

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:

$$P_1 = (1 - \eta)\eta P. \quad (2.3.111)$$

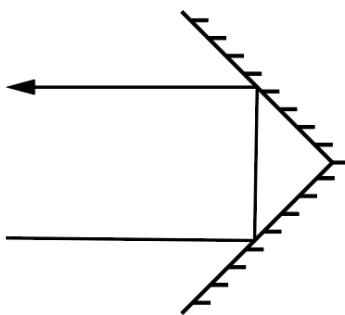


Рис. 2.3.18

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

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

Погрешность 0,1 мДж.

Ответ: $(8,0 \pm 0,1) \text{ мДж.}$

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.

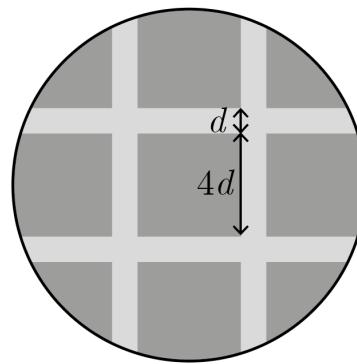


Рис. 2.3.19

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.

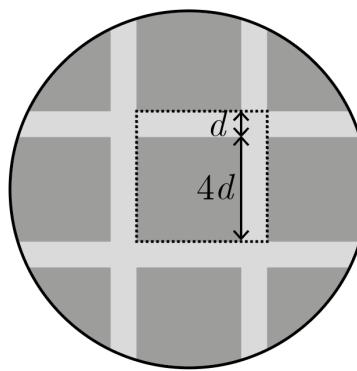


Рис. 2.3.20

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

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

Тогда искомая плотность окончательно равна

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

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 поршины 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/с². На рисунке точный масштаб не сохранен.

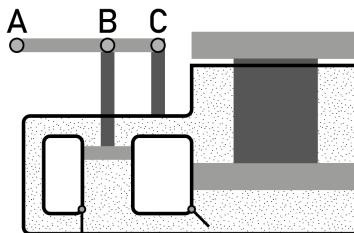


Рис. 2.3.21

Решение

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

$$\frac{mg}{F_1} = \frac{\pi R^2}{\pi r^2} \Rightarrow F_1 = mg \frac{r^2}{R^2}, \quad (2.3.115)$$

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

$$\frac{F_1}{F} = \frac{AC}{BC} = \frac{AB + BC}{BC} = 1 + \frac{AB}{BC}. \quad (2.3.116)$$

Совместив эти уравнения, получим окончательно

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

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

$$P = I^2 R, \quad (2.3.118)$$

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

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

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

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

Эту систему можно решить, вычитая второе уравнение из первого

$$\frac{\Delta P_1 + \Delta P_2}{P_0} = 4x \Rightarrow x = \frac{\Delta P_1 + \Delta P_2}{4P_0} \quad (2.3.121)$$

и подставляя результат в любое уравнение системы (2.3.120)

$$\chi + \frac{\Delta P_1}{P_0} = \chi + \frac{\Delta P_1 + \Delta P_2}{2P_0} + \frac{(\Delta P_1 + \Delta P_2)^2}{16P_0^2}. \quad (2.3.122)$$

Домножим на $16P_0^2$

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

и выразим окончательно

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

Погрешность 1 Вт.

Ответ: 100 ± 1 Вт.

2.3.8. Четвертая волна. Задачи 10–11 класса

Задачи четвертой волны предметного тура по физике за 10–11 класс открыты для решения. Соревнование доступно на платформе Яндекс.Конкурс: <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

$$a = \frac{v^2}{R} = \omega^2 R, \quad (2.3.125)$$

где R — радиус траектории, v — линейная скорость, ω — угловая скорость. Пере-
множив эти выражения, получим

$$a^2 = \frac{v^2}{R} \omega^2 R \Rightarrow a = v\omega. \quad (2.3.126)$$

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

$$\frac{a_A}{a_B} = \frac{v_A}{v_B} \cdot \frac{\omega_A}{\omega_B} = \frac{1}{2} \cdot \frac{3}{1} = 1,5. \quad (2.3.127)$$

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 $q = 20 \text{ мККл}$. 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 $E = 2,5 \text{ кВ/м}$, сила натяжения нити после установления равновесия оказывается равна $T = 130 \text{ мН}$. Ускорение свободного падения $g = 9,8 \text{ м/с}^2$.

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

$$q\vec{E} + m\vec{g} + \vec{T} = \vec{0}. \quad (2.3.128)$$

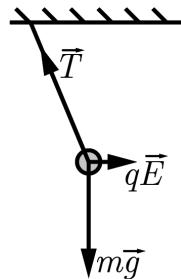


Рис. 2.3.22

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

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

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, ав-

томобиля с автопилотом равна среднему арифметическому координат x_1 идущего впереди и x_2 идущего позади

$$x = \frac{x_1 + x_2}{2}. \quad (2.3.130)$$

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

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

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

$$a_x = \frac{a_{x1} + a_{x2}}{2}. \quad (2.3.132)$$

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

$$a_{x2} = 2a_x - a_{x1}. \quad (2.3.133)$$

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, наоборот, ускоряется (имеет положительную проекцию). Тогда напрямую из (2.3.133) получим

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

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

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

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

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

Совмешая эти равенства, получим окончательно

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

Погрешность $0,02 \text{ м/с}^2$.

Ответ: $(1,96 \pm 0,02) \text{ м/с}^2$.

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

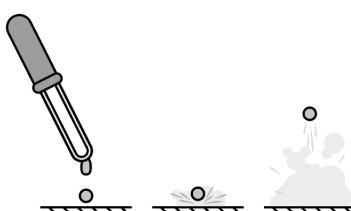


Рис. 2.3.23

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 этой части энергии

$$\Delta U = (1 - \eta)Q = Lm\alpha, \quad (2.3.138)$$

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

$$Q = \frac{Lm\alpha}{1 - \eta}. \quad (2.3.139)$$

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

$$A = \eta Q = (1 - \alpha)mgh. \quad (2.3.140)$$

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

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

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/66690/enter/>.

Задача 2.4.1. Программирование микроконтроллеров. Поворотные механизмы (25 баллов)

Темы: Arduino Nano, микроконтроллеры, Wokwi, джойстик.

Условие

Дана схема, состоящая из контроллера Arduino Nano, сервомотора, джойстика, RGB-светодиода и зуммера. Схема по условию должна работать следующим образом:

- При наклоне джойстика вперед сервопривод поворачивается направо на 90°, загорается зеленый светодиод.
- При наклоне джойстика назад сервопривод поворачивается налево на 90°, загорается синий светодиод.
- При наклоне влево/вправо загорается красный светодиод.

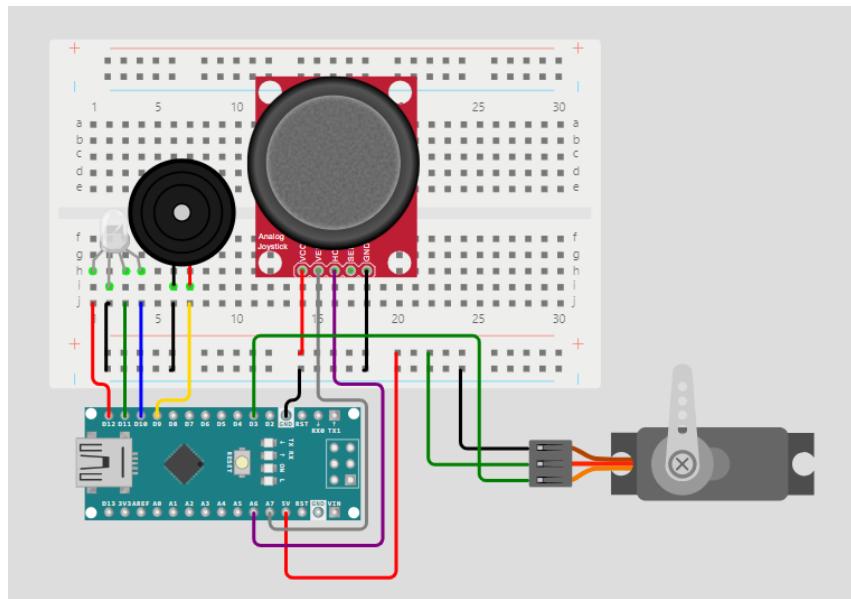


Рис. 2.4.1. Электрическая схема

Ниже приведен код программы, который необходимо исправить так, чтобы он работал согласно условию.

C++

```

1  #include <Servo.h>
2
3  #define LED_RED      12 // светодиод красный
4  #define LED_GREEN    11 // светодиод зеленый
5  #define LED_BLUE     10 // светодиод желтый
6
7  #define BUZZ_PIN      9 // зуммер
8  #define HOR_JOY       A6 // джойстик ось X
9  #define VERT_JOY      A7 // джойстик ось Y
10
11 #define SERVO_PIN    3 // серво
12
13 Servo servo;
14
15 unsigned long time_passed, time_reset;
16
17 void setup() {
18     pinMode (LED_RED, OUTPUT);
19     pinMode (LED_GREEN, OUTPUT);
20     pinMode (LED_BLUE, OUTPUT);
21
22     pinMode (BUZZ_PIN, OUTPUT);
23     servo.attach(SERVO_PIN);
24     servo.write(90);
25 }
26
27 void loop() {
28     int horizontal = analogRead(HOR_JOY);
29     int vertical = analogRead(VERT_JOY);
30
31     if (your_answer) {
32         servo.write(your_answer);
33         digitalWrite(LED_RED, LOW);
34         digitalWrite(LED_GREEN, HIGH);
35         digitalWrite(LED_BLUE, LOW);
36     }
37     if (your_answer) {
38         servo.write(your_answer);
39         digitalWrite(LED_RED, LOW);
40         digitalWrite(LED_GREEN, LOW);
41         digitalWrite(LED_BLUE, HIGH);
42     }
43     if (your_answer){
44         servo.write(90);
45         digitalWrite(LED_RED, LOW);
46         digitalWrite(LED_GREEN, LOW);
47         digitalWrite(LED_BLUE, LOW);
48     }
49
50     if (your_answer) {
51         digitalWrite(LED_RED, HIGH);
52         digitalWrite(LED_GREEN, LOW);
53         digitalWrite(LED_BLUE, LOW);
54     }
55     // Подзадача 1.2
56 }
```

Подзадача 1 (13 баллов)

Найдите ошибку в коде и вставьте нужные строки в области кода вида `your_answer`. Ответом является шесть цифр, записанных в верной последовательности (например, 123456).

Варианты ответа:

1. `horizontal != 512`
2. `vertical == 512`
3. `180`
4. `vertical > 512`
5. `vertical < 512`
6. `0`

Подзадача 2 (12 баллов)

Дополните программу таким образом, чтобы при отсутствии действия больше 10 с светодиод начал мигать желтым цветом, а зуммер начал подавать звуковой сигнал в слышимом диапазоне (от 20 Гц до 20 кГц).

Выберите один из четырех предложенных вариантов реализации, который нужно добавить в программу после комментария // Подзадача 1.2.

C++

```

1. 1 time_passed = millis() - time_reset;
2  if (horizontal != 512 and vertical != 512) {
3    time_reset = millis();
4  }
5
6  if ((10000 < time_passed)) {
7    digitalWrite(LED_RED, LOW);
8    digitalWrite(LED_GREEN, LOW);
9    digitalWrite(LED_BLUE, LOW);
10   delay(100);
11   digitalWrite(LED_RED, HIGH);
12   digitalWrite(LED_GREEN, HIGH);
13   digitalWrite(LED_BLUE, LOW);
14   delay(100);
15
16   tone(BUZZ_PIN, 250, 100);
17 }
```

C++

```

2. 1 time_passed = millis() - time_reset;
2  if (horizontal != 512 or vertical != 512) {
3    time_reset = millis();
4  }
5
6  if ((10000 < time_passed)) {
7    digitalWrite(LED_RED, LOW);
8    digitalWrite(LED_GREEN, LOW);
9    digitalWrite(LED_BLUE, LOW);
10   delay(100);
```

```

11  digitalWrite(LED_RED, HIGH);
12  digitalWrite(LED_GREEN, HIGH);
13  digitalWrite(LED_BLUE, LOW);
14  delay(100);
15
16  tone(BUZZ_PIN, 250, 100);
17 }
```

C++

3.

```

1  time_passed = millis() - time_reset;
2  if (horizontal == 512 or vertical == 512) {
3    time_reset = millis();
4  }
5
6  if ((10000 <= time_passed)) {
7    digitalWrite(LED_RED, LOW);
8    digitalWrite(LED_GREEN, LOW);
9    digitalWrite(LED_BLUE, LOW);
10   delay(100);
11   digitalWrite(LED_RED, HIGH);
12   digitalWrite(LED_GREEN, HIGH);
13   digitalWrite(LED_BLUE, LOW);
14   delay(100);
15
16   tone(BUZZ_PIN, 250, 100);
17 }
```

C++

4.

```

1  time_passed = millis() - time_reset;
2  if (horizontal != 512 or vertical == 512){
3    time_reset = millis();
4  }
5
6  if ((10000 <= time_passed)) {
7    digitalWrite(LED_RED, LOW);
8    digitalWrite(LED_GREEN, LOW);
9    digitalWrite(LED_BLUE, LOW);
10   delay(100);
11   digitalWrite(LED_RED, HIGH);
12   digitalWrite(LED_GREEN, HIGH);
13   digitalWrite(LED_BLUE, LOW);
14   delay(100);
15
16   tone(BUZZ_PIN, 250, 100);
17 }
```

Ответ:

Подзадача 1: 435621.

Подзадача 2: 2.

Задача 2.4.2. Конструирование. Определение грузоподъемности (25 баллов)

Темы: THPA, тяга, плавучесть.

Условие

Дан ТНПА с нулевой плавучестью. К сожалению, вертикальный движитель неисправен и требует замены. Перед установкой нового движителя было принято решение испытать его на лабораторном стенде.

В результате испытаний был зафиксирован вес на свободном плече стенда: 54 Н.

Длина свободного плеча: 10 см, длина плеча с движителем 75 см.

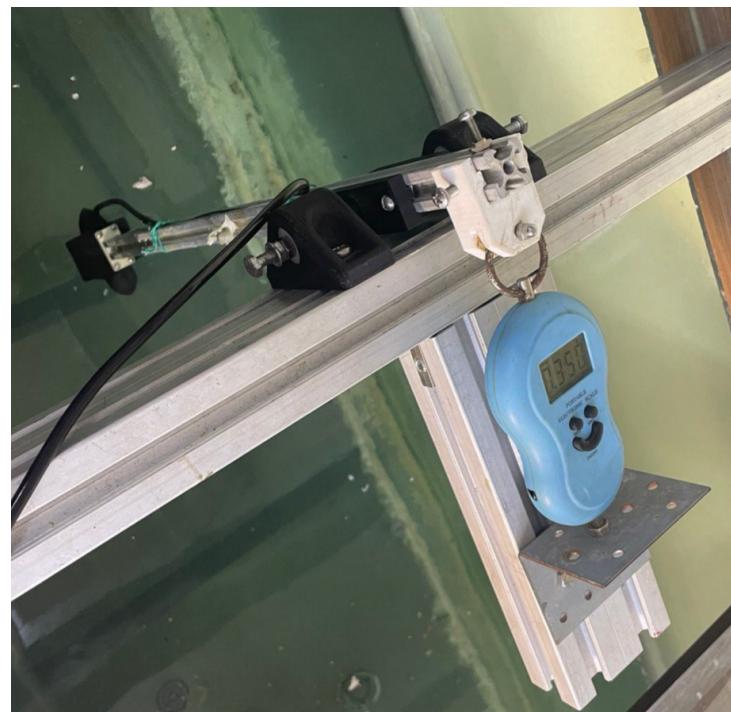


Рис. 2.4.2. Вид лабораторной установки

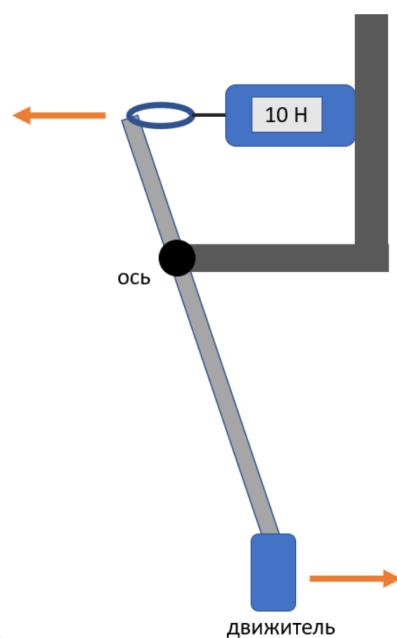


Рис. 2.4.3. Схема лабораторной установки

Подзадача 1 (10 баллов)

Определите тягу движителя. Ответ укажите в ньютонах с точностью до десятых.

Подзадача 2 (15 баллов)

Рассчитайте тягу движителя по показаниям лабораторного стенда. Определите, хватит ли тяги такого движителя, установленного на аппарат с нулевой плавучестью, чтобы поднять куб со стороной 10 см и весом 15 Н. В ответе укажите сумму выталкивающих сил.

Решение

Из условия равновесия рычага:

$$F_{\text{д}} = 54 \cdot \frac{10}{75} = 7,2 \text{ Н.}$$

Сила выталкивания куба:

$$F = 1000 \cdot 9,8 \cdot 0,001 = 9,8 \text{ Н.}$$

Сумма выталкивающих сил:

$$F_{\text{в}} = 9,8 + 7,2 = 17 \text{ Н}$$

— это больше веса 15 Н, поэтому аппарат сможет поднять такой груз со старым движителем.

Ответ:

Подзадача 1: 7,2 Н.

Подзадача 2: 17 Н.

Задача 2.4.3. Программирование. Расчет расстояния (25 баллов)

Темы: Python, программирование, компьютерное зрение.

Условие

Программа получает изображение, на котором находятся два желтых прямоугольника и один черный квадрат. Необходимо вычислить расстояние между прямоугольниками в сантиметрах, если известно, что сторона черного квадрата фиксирована и составляет 10 см.

Ниже представлена реализация программы, однако часть кода пропала, а в одной строке обнаружена ошибка. Устраните все недочеты и восстановите программу.

Для тестирования программы авторы используют три изображения. Изображения можно скачать по ссылке: https://disk.yandex.ru/d/Plmqol_Xpn80GQ/изображения.

Пример изображения на рис. 2.4.4.



Рис. 2.4.4

Ниже приведен код программы, который необходимо исправить так, чтобы он работал согласно условию.

Python

```

1  import cv2
2
3  img = cv2.imread("slide1.jpg")
4
5  yellow_color = ((0, 50, 50), (40, 255, 255))
6  black_color = ((0, 0, 0), (180, 255, 30))
7  yellow_points = []
8  cube_width = 0
9
10 def find_contours(img, color):
11     img_hsv = cv2.cvtColor(img, cv2.COLOR_BGR2HSV)
12     img_mask = cv2.inRange(img_hsv, color[0], color[1])
13     contours, _ = cv2.findContours(img_mask, cv2.RETR_EXTERNAL,
14         → cv2.CHAIN_APPROX_SIMPLE)
15     return contours
16
17 yellow_cnt = find_contours(img, yellow_color)
18 black_cnt = find_contours(img, black_color)
19
20 for cnt in black_cnt:
21     x, y, w, h = cv2.boundingRect(cnt)
22     cube_width = your_answer
23
24 for cnt in yellow_cnt:
25     x, y, w, h = cv2.boundingRect(cnt)
26     yellow_points.append(x)
27     yellow_points.append(your_answer)
28
29 yellow_points.sort()
30 dist = your_answer - your_answer
31 result = dist / (insert_here / 10)
32
33 print(round(result))
  
```

Формат выходных данных

Выходные данные представляют собой строку, которая содержит целое число, которое равно расстоянию между прямоугольниками в сантиметрах.

Подзадача 1 (13 баллов)

Расположите строки в правильном порядке (задача на сопоставление).

1. `yellow_points[1]`
2. `x + w`
3. `yellow_points[2]`
4. `(x + w) - x`

Подзадача 2 (12 баллов)

Замените неправильный параметр в строке на верный (задача с вводом текста).

Python

```
1 result = dist / (insert_here / 10)
```

Ответ:

Подзадача 1: 4, 2, 3, 1.

Подзадача 2: `cube_width`.

Задача 2.4.4. Электрические схемы (25 баллов)

Темы: электрические цепи, падение напряжения, электроника.

Условие

Подзадача 1 (7 баллов)

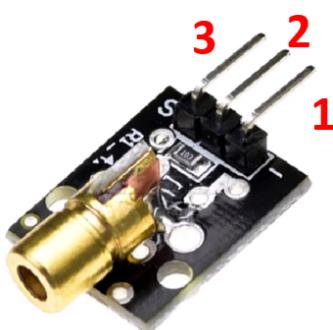


Рис. 2.4.5. Лазерный модуль KY-008

Иван хочет разработать герметичный дальномер, который может определять размеры повреждений. Для этого он решил использовать лазерный модуль KY-008 (рис. 2.4.5), фоторезистор и плату Arduino Nano (рис. 2.4.6).

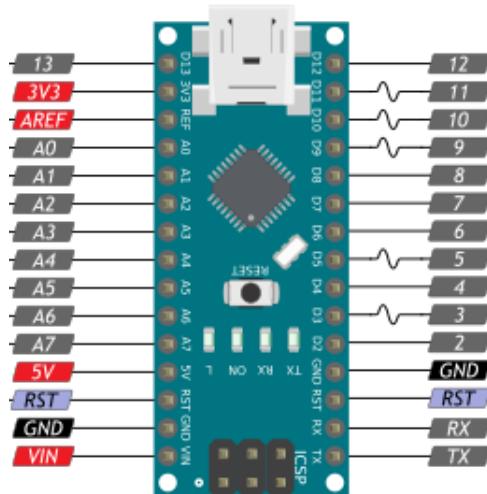


Рис. 2.4.6. Arduino Nano

Ознакомьтесь с инструкцией к модулю и ответьте на следующий вопрос.

К каким пинам платы Arduino Nano Иван должен подключить лазерный модуль?

Подключите выводы 1, 2, 3 лазерного модуля к плате, выбрав правильные варианты. Ответом является последовательность из трех цифр.

1. Tx,
2. D6,
3. RST,
4. GND,
5. 5V,
6. VIN.

Подзадача 2 (7 баллов)

Для обеспечения работы всех систем бортовой части подводного аппарата необходимо правильно подбирать провода для изготовления кабель-троса. Для этого необходимо знать, какие потери напряжения ожидают пользователя при использовании кабеля того или иного сечения.

Ученники провели исследования своего подводного устройства и выяснили, что при подключении подводного устройства к батарее 12 В через кабель-трос на движитель приходит только 10,68 В.

Известно, что для питания кабеля используются два одножильных провода по 20 м неизвестного сечения, сила тока в цепи 4 А.

Рассчитайте потери напряжения на кабеле. Ответ округлите до сотых.

Подзадача 3 (11 баллов)

Определите, провода какого сечения были использованы по стандарту AWG. Для определения характеристик провода нужно воспользоваться таблицей: <https://anlan.ru/table-awg>

Решение

Тогда потери напряжения на кабеле:

$$Uk = 12 - 10,68 = 1,32 \text{ В.}$$

Сопротивление кабеля:

$$R = \frac{1,32}{4} = 0,33 \text{ Ом.}$$

Погонное сопротивление:

$$Rp = \frac{0,33 \text{ Ом}}{0,04 \text{ км}} = 8,25 \text{ Ом/км.}$$

По справочнику стандарта AWG определяем, что данным погонным сопротивлением обладает кабель сечением 14 AWG.

Ответ:

Подзадача 1: 452.

Подзадача 2: 1,32 В.

Подзадача 3: 14 AWG.

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 помочь 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 второго этапа являются декомпозицией задач заключительного этапа. Их решение позволяет ознакомиться с основными технологиями диагностики судна с помощью подводных аппаратов и мультисенсорных зондов, которым и посвящен заключительный этап.

Участники команды при решении задач подробнее знакомятся со своими ролями и тем, что им предстоит делать в рамках заключительного этапа:

1. программисту — программировать в MUR IDE движение робота, регуляторы, алгоритмы компьютерного зрения;
2. электронщику — выполнять пайку и электромонтаж при подготовке подводного аппарата к выполнению миссии;
3. программисту микроконтроллеров — программировать в Arduino IDE и настраивать работу микроконтроллера;
4. конструктору — проектировать в САПР, разрабатывать чертежи, работать с формообразующим инструментом и готовить конструкторскую документацию.

Во втором этапе представленный пакет задач состоит из индивидуальных задач, проверяющих личные навыки участников команды согласно их компетенциям (программирование роботов в симуляторе, программирование микроконтроллеров, электроника, конструирование), и командной задачи на разработку коммерческого предложения, которая требует совместной работы.

Участники должны наладить систему планирования и коммуникации внутри команды, правильно распределять задачи, согласно своим ролям и возможностям.

- Комплект задач состоит из 5 задач.
- Общая сумма баллов: 300 баллов.

Компетенции, необходимые для решения задач второго этапа:

- программирование Python,
- программирование в симуляторе MUR IDE,
- компьютерное зрение,
- регуляторы,
- Arduino,
- Tinkercad,
- схемотехника,
- проектирование электрических плат,
- работа с ГОСТ,
- проектирование в САПР,
- разработка конструкторской документации.

3.2.1. Индивидуальные задачи

Комплект индивидуальных задач состоит из четырех задач. Три задания размещены на платформе Яндекс.Контест и предполагают ручную проверку.

Задача для программиста компьютерного зрения размещена на платформе MUR Contest. Данное задание оценивается автоматически.

Индивидуальные задачи второго этапа инженерного тура открыты для решения. Соревнование доступно на платформе Яндекс.Контест: <https://contest.yandex.ru/contest/69892/enter/>.

Задача 3.2.1.1. Конструирование (50 баллов)

Темы: конструирование, САПР, моделирование, герметичность.

Условие

Для проведения комплекса работ по исследованию дна морского судна компания N решила использовать телеуправляемый необитаемый подводный аппарат (ТНПА). Однако установленная на нем аналоговая камера не подходила под задачи исследований, поэтому инженерами было принято решение заменить эту камеру на цифровую, а заодно заменить сервопривод.

Задача участников: спроектировать герметичный корпус камеры, исходя из заданных характеристик используемых материалов и компонентов, разработать крепление камеры к сервоприводу.

Герметичный корпус состоит:

- из акриловой трубы (гильзы);
- из крышек;
- из системы крепления герметичной камеры к раме подводного аппарата (отверстия, крепежные элементы, дополнительные детали);
- из узла крепления камеры к сервоприводу и всей конструкции внутри корпуса;
- из устройства герметичного вывода кабеля через крышку наружу.

Для разработки прототипа участникам предложены следующие компоненты:

- Материал для гильзы: акриловая труба из ассортимента: <https://disk.yandex.ru/i/TDfQpo4ZaOdq2w>.
- Цифровая камера, 1 шт.: <https://clck.ru/3Fre4W>.
- Сервопривод, 1 шт.: <https://hitecrcd.com/hs-85bb-premium-micro-servo/>.
- Материал для крышек: алюминиевый пруток АМг3 (ГОСТ 4784-97) на выбор из сортамента \varnothing 45–95 мм с шагом диаметра 10 мм.
- Набор необходимых уплотнительных колец согласно ГОСТ 9833-73: <https://files.stroyinf.ru/Index2/1/4294820/4294820621.htm>.

Характеристики камеры

- HBVCAM-1716-130.

- Габариты: $38 \times 38 \times 31$ мм.
- Угол обзора 130° .



Рис. 3.2.1

Характеристики сервопривода

- HS-85BB.
- Габариты: $39 \times 25,5 \times 13$ мм.
- На валу установлена крестовина.



Рис. 3.2.2

Требования к конструкции

1. Для герметизации корпуса необходимо использовать уплотнительное кольцо сечением **2,5 мм**. Минимально допустимая толщина стенок гильзы — **3 мм**.
2. Герметичная камера должна крепиться к листовому материалу толщиной **10 мм**. Предполагается, что герметичная камера должна устанавливаться в паз в листовой панели так, как это показано на рис. 3.2.3.

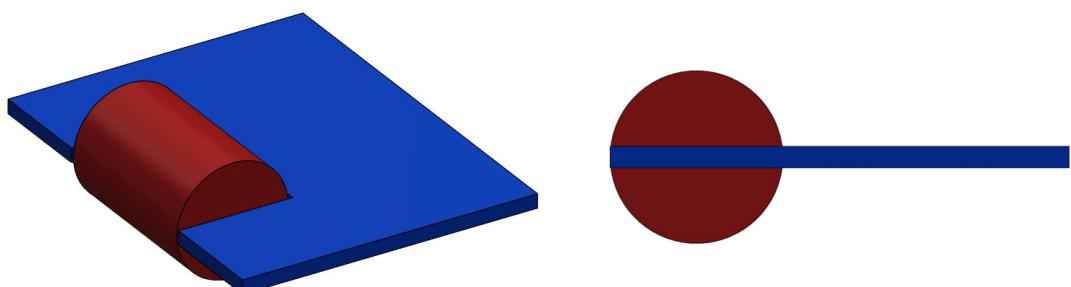


Рис. 3.2.3. Установка герметичной камеры

3. Камера должна свободно проворачиваться внутри гильзы на 360°.
4. Должен быть предусмотрен способ крепления крышек к гильзе, препятствующий осевому смещению относительно друг друга. Это необходимо для предотвращения разгерметизации корпуса камеры из-за возникновения разности давления внутри и снаружи корпуса.
5. Должен быть предусмотрен способ герметичного вывода кабеля камеры через крышку наружу, диаметр кабеля — 5,5 мм.
6. Для крепления камеры к сервоприводу нужно использовать крестовину, указанную в чертеже сервопривода.
7. Остальные требования к конструкции указаны в листе оценки. Участники должны внимательно ознакомиться с листом оценки перед проектированием.

Лист оценки: https://disk.yandex.ru/i/_MomKsinWvFXXA.

Решение

1. Результатом работы является:
 - модель корпуса герметичной камеры с внутренним наполнением (камерой, сервоприводом) и ответной частью в виде листового материала толщиной 10 мм с пазом под камеру;
 - сборочный чертеж (без изображения ответной части);
 - чертеж гильзы;
 - чертежи крышек;
 - чертежи дополнительных деталей, если такие имеются;
 - спецификация;
 - пояснительная записка.
2. Решение задачи конструктора должно соответствовать критериям, представленным в листе оценки.
3. Чертежи необходимо выполнять согласно ГОСТ 2.109-73: <https://filesstroyinf.ru/Data2/1/4294852/4294852138.pdf>.
4. Спецификация и пояснительная записка должны быть выполнены согласно ГОСТ Р 2.106-2019: <https://filesstroyinf.ru/Data/708/70838.pdf>.
5. Оценка сборочного чертежа и спецификации производится по упрощенным критериям, так что ориентироваться прежде всего необходимо на лист оценки.
6. Для пояснительной записи предлагается упрощенный вариант содержания:
 - введение (с указанием, на основании каких документов разработан проект);
 - наименование и область применения проектируемого изделия;
 - описание и обоснование способа крепления камеры с сервоприводом к крышке и способа крепления герметичной камеры к листовой панели;
 - расчеты для уплотнительных колец;
 - обоснование выбора допусков согласно ГОСТ;
 - описание способа герметичного вывода кабеля из корпуса;
 - планируемый способ изготовления каждой детали (кроме стандартных изделий).

7. Решение задачи должно быть предоставлено в виде двух следующих файлов:
- pdf-файл с чертежами гильзы, крышек, дополнительных деталей (если такие имеются), сборочным чертежом, спецификацией и пояснительной запиской;
 - step-файл с моделью.

В качестве возможного варианта решения приводится работа команды, набравшая максимальное количество баллов (47 баллов из 50 баллов):

<https://disk.yandex.ru/d/-OunV-JWsM1zmw/конструирование>.

Задача 3.2.1.2. Программирование в симуляторе (50 баллов)

Темы: Python, компьютерное зрение, регуляторы, MUR IDE.

Для выполнения задания необходимо:

1. Скачать и установить среду разработки MUR IDE для программирования виртуального автономного подводного робота: <https://murproject.com/#download>.
2. После установки на рабочем столе должен появиться ярлык MUR IDE. Запустить MUR IDE.
3. Для начала работы в симуляторе перевести режим работы IDE в Local. Для этого следует нажать на кнопку с иконкой ракеты и надписью **Remote** в левом верхнем углу. Цвет кнопки станет синим и надпись изменится на **Local**.

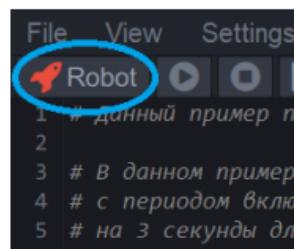


Рис. 3.2.4

4. Запустить симулятор нажатием кнопки с изображением жука в правом верхнем углу. Откроется окно симулятора с черным экраном.

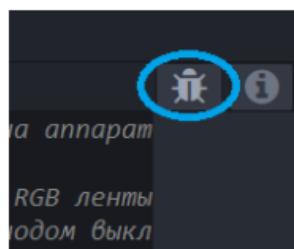


Рис. 3.2.5

5. Скачать сцену по ссылке. Можно разместить ее в любом удобном каталоге.
6. В запущенном симуляторе перейти в меню **Scene** → **Open**. В появившемся диалоговом окне выбрать сцену.

7. Все готово, можно начать программировать виртуальный аппарат на языке программирования Python.
8. В качестве решения задачи необходимо отправить файл с кодом в формате *.py. Для этого в меню следует выбрать **File → File save as**.

Условие

При проведении работ по осмотру дна морских судов и мелкому ремонту все чаще компании прибегают к помощи подводных аппаратов.

Необходимо выполнить работы по диагностике состояния судна, запрограммировав аппарат на обнаружение повреждений и прочих объектов согласно заданию.

Описание сцены

В сцене представлен корабль, корпус которого окрашен в красный и серый цвета. Для удобства ориентирования вокруг корабля на дне нарисована фиолетовая полоса. Начальное положение аппарата: напротив кормы корабля у винтов.

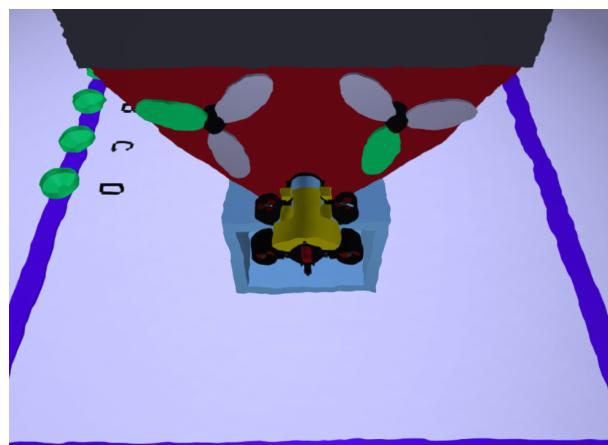


Рис. 3.2.6. Начальное положение аппарата

Задание

1. Дефектоскопия (10 баллов).

- На правом борту судна расположены повреждения оранжевого цвета. Необходимо определить порядковый номер самого крупного повреждения.
- Порядковый номер повреждения определен. Это означает, что **аппарат выстрелил в розовую мишень (находится в конце ряда повреждений) количеством торпед, равным порядковому номеру мишени.**

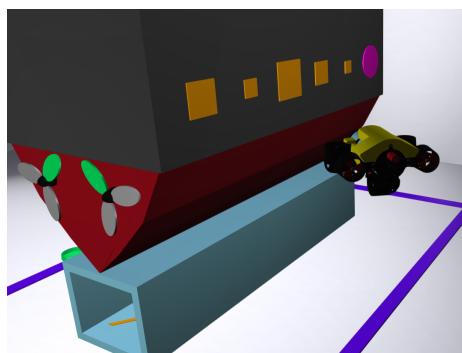


Рис. 3.2.7. Дефектоскопия

2. Очистка винтов (10 баллов).

- На корме расположены два рулевых винта серого цвета. Одна из лопастей каждого винта покрылась обрастаниями (такая лопасть будет окрашена в зеленый цвет).
- Необходимо выполнить очистку винтов от обрастаний.
- Винт очищен. Это означает, что **аппарат коснулся зеленой лопасти своей передней частью (колбой)**.
- За касание чистой лопасти (лопасть серого цвета) предусмотрен штраф. Подробнее — в разделе «**Критерии оценивания**».
- При проверке данной задачи при успешном касании зеленой лопасти ее цвет сменится на серый.

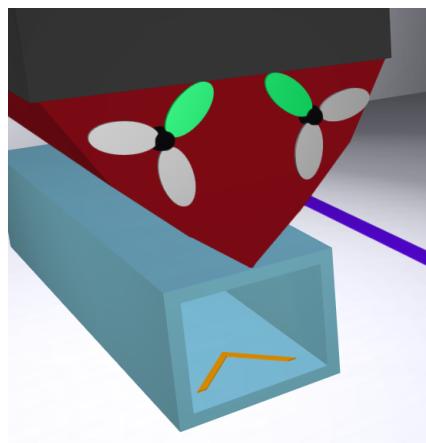


Рис. 3.2.8. Очистка винтов

3. Определение размеров трещины (15 баллов).

- На левом борту судна обнаружена трещина белого цвета.
- Над трещиной расположен желтый калибровочный квадрат с известной длиной стороны одна единица.
- Необходимо выяснить, сколько единиц составляет длина трещины и указать ее.

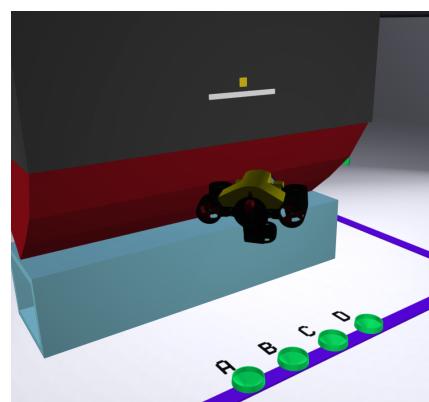


Рис. 3.2.9. Определение размеров трещины

- Для указания длины трещины рядом расположены четыре корзины зеленого цвета. Рядом с каждой корзиной указана буква, обозначающая разряд. Размер необходимо закодировать с применением двоичного кода, сбросив маркеры (см. таблицу кодировки 3.2.1).
- Отсутствие маркера в корзине приравнивается к 0.
- Наличие маркера в корзине приравнивается к 1.
- Длина трещины может меняться от 1 до 9 и всегда является целым числом.

Таблица 3.2.1. Таблица кодировки

| Длина | Корзины | | | |
|-------|---------|---|---|---|
| | A | B | C | D |
| 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 |
| 2 | 0 | 0 | 1 | 0 |
| 3 | 0 | 0 | 1 | 1 |
| 4 | 0 | 1 | 0 | 0 |
| 5 | 0 | 1 | 0 | 1 |
| 6 | 0 | 1 | 1 | 0 |
| 7 | 0 | 1 | 1 | 1 |
| 8 | 1 | 0 | 0 | 0 |
| 9 | 1 | 0 | 0 | 1 |

4. Исследование корпуса (10 баллов).

Необходимо исследовать дно корабля под килем. Задание выполнено верно, если аппарат зашел со стороны винтов и вышел со стороны носовой части. Вход в коридор со стороны кормы обозначен оранжевой стрелкой.

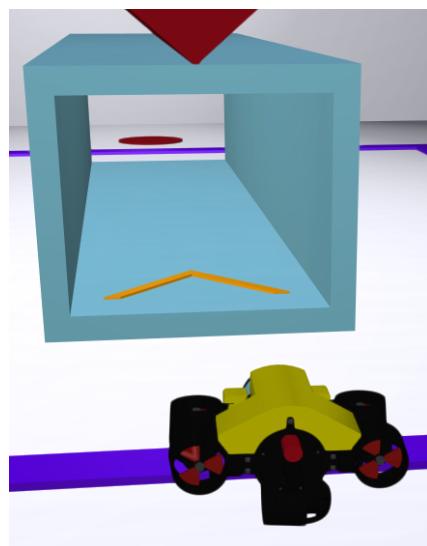


Рис. 3.2.10. Исследование корпуса

5. **Всплытие (10 баллов).**

После завершения миссии аппарат должен всплыть со стороны носовой части над красным кругом.

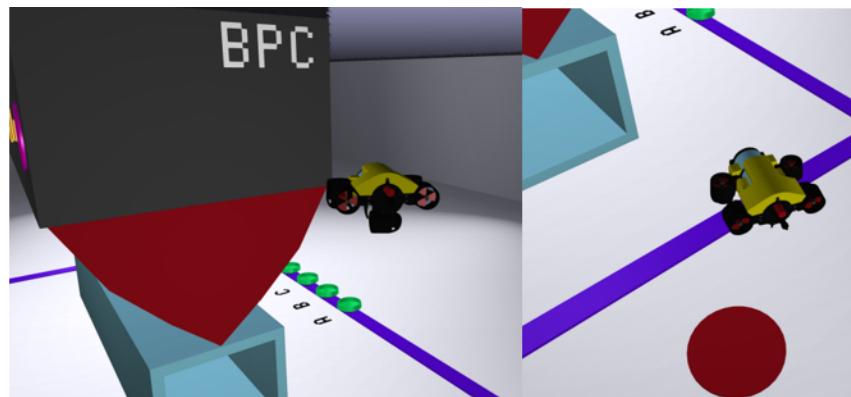


Рис. 3.2.11. Область всплытия

Требования к выполнению задания

- Время выполнения задания (попытка) — не более **5 мин.**
- Попытка завершается в тот момент, как робот всплыл на поверхность. Всплытие на поверхность означает, что своей верхней частью он пересек поверхность воды. В этом случае время останавливается, и в зачет идут те баллы, которые робот успел набрать до всплытия.
- Попытка завершается, если истекло максимальное время на попытку (4 мин). В этом случае время выполнения попытки устанавливается 4 мин, и в зачет идут те баллы, которые робот успел набрать до остановки времени.
- Попытка останавливается в случае завершения кода.

Критерии оценивания

Баллы за прохождение начисляются следующим образом.

| Задача | Критерий | Балл |
|--------|---|------|
| 1 | Определен самый большой дефект (количество выстрелов в мишень совпадает с порядковым номером объекта) | 10 |
| 2 | Винты очищены (по 5 за каждый) | 10 |
| 3 | Определен размер трещины (15 — верно, 5 — с ошибкой ± 1) | 15 |
| 4 | Обследование выполнено | 10 |
| 5 | Всплытие | 5 |
| 6 | Штраф за маркер вне корзины | -2 |
| 7 | Штраф за снаряд, выпущенный не в мишень | -2 |
| 8 | Штраф за касание неверной (серой) лопасти | -2 |

Подготовка и тестирование

Для тестирования участникам доступны 3 сцены. Контрольные сцены отличаются от тестовых. Скачать сцены для тестирования можно по ссылке: https://disk.yandex.ru/d/-OunV-JWsM1zmw/test_scenes.

Размеры трещин в каждой схеме:

- `scene_1.xml` — 9;
- `scene_2.xml` — 5;
- `scene_3.xml` — 3.

Что может меняться

- Размер квадратов-повреждений.
- Длина трещины.
- Расположение обрастаний на винтах.
- Центр трещины всегда совпадает с центром калибровочного квадрата. При этом положение калибровочного квадрата не меняется.

Примечание. Положение корзин не меняется.

Порядок оценки решений и работа системы автоматической проверки

- Решением задачи является файл в формате `*.py`, который загружается на платформу MUR Contest и оценивается автоматически.
- В MUR contest будут доступны два соревнования: НТО (тестовые) и НТО (финальные).
- В НТО (тестовые) можно выполнять проверку своего решения на тестовых сценах и получать результаты. Количество попыток: 10.
- В НТО (финальные) загружается итоговое решение. Это решение проверяется на контрольных сценах после завершения приема работ.
- С инструкциями по работе с системой можно ознакомиться в личном кабинете участника внутри соревнования.
- Участники могут заменять файл итоговой программы до завершения приема работ.
- Заключительная проверка выполняется после завершения приема работ второ-

го этапа. **Проверку проходит последний файл, присланный участником.**

- Задача проверяются на трех сценах, которые отличаются от предоставленных.
- Сцены различаются расположением объектов, их цветом.
- В зачет идет **худший** результат из трех попыток.
- Все работы проверяются на плагиат.

Ответ: ссылка на вариант решения от разработчиков с комментариями:

<https://disk.yandex.ru/d/-OunV-JWsM1zmw/симулятор>.

Задача 3.2.1.3. Программирование микроконтроллеров (50 баллов)

Темы: микроконтроллеры, SimulIDE, датчики, Arduino Uno, Arduino Nano.

Условие

Для определения глубины отверстия в конструкциях, установленных под водой, для ТНПА планируется реализовать специальный модуль. Для тестирования и отладки предполагаемого функционала была собрана схема устройства в среде SimulIDE.

Модуль представляет собой герметичное устройство на основе Arduino Nano с ультразвуковым датчиком и сервомотором.

Все данные передаются на пульт управления на основе Arduino Uno с подключенными джойстиком, кнопкой, дисплеем, RGB-светодиодом.

Модель устройства в SimulIDE можно скачать по ссылке: <https://disk.yandex.ru/d/G4f61QeTB0YogA>.

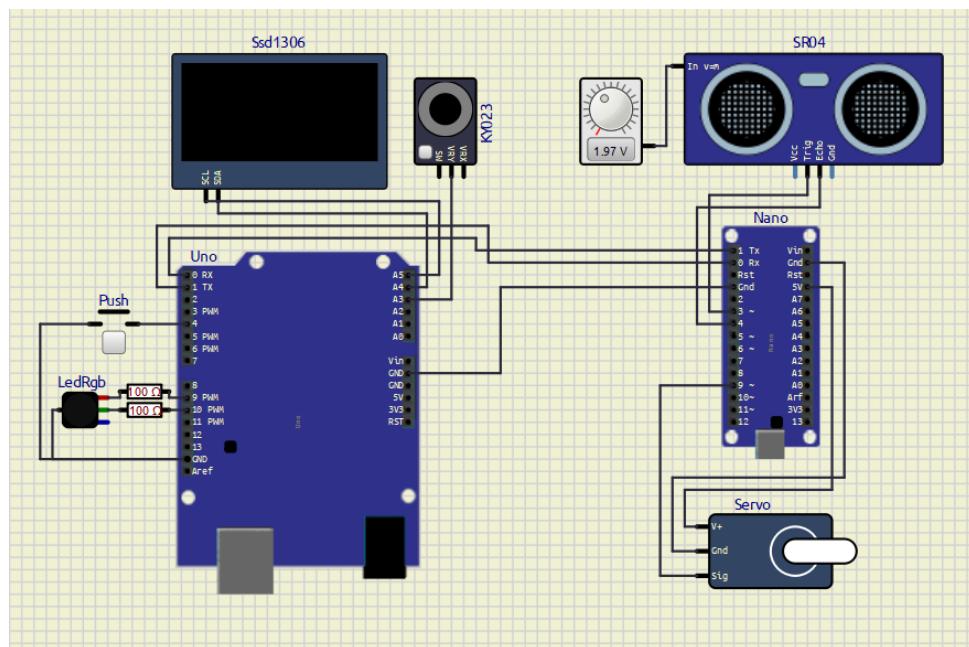


Рис. 3.2.12. Общая схема модели устройства

Устройство должно функционировать следующим образом:

- Движение устройства (вправо/влево):
 - джойстик вперед — поворот сервы до положения 0;
 - джойстик назад — поворот сервы до положения 180;
 - при достижении крайних значений сервомотор останавливается.
- Нажатие кнопки — регистрация расстояния до препятствия:
 - 1 нажатие — сообщение START появляется на второй секунде;
 - 2 нажатие — фиксация S1;
 - 3 нажатие — фиксация S2.
- Модуль измеряет расстояния до внешней стенки полости, фиксирует значение S1 и измеряет расстояние до внутренней стенки полости, фиксирует значение S2. Далее определяется глубина полости (**S2 – S1**).
- Расстояния и итоговая глубина выводятся на дисплей.
- Если расстояние в пределах нормы, то RGB-светодиод светится зеленым.
- Если расстояние ниже нормы, то RGB-светодиод светится красным.
- Если расстояние выше нормы, то RGB-светодиод светится желтым.
- При следующем нажатии на кнопку цикл измерений повторяется заново.

Примечания

Нормальным считается измерение в пределах 30–70 см.

Допустимая погрешность измерений УЗ-датчика: ± 3 см.

Диапазон значений УЗ-датчика

Изменение расстояния моделируется в SimulIDE при помощи источника питания.

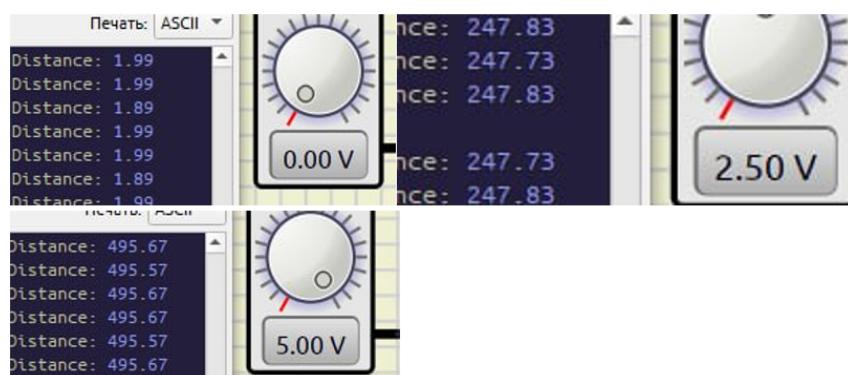


Рис. 3.2.13

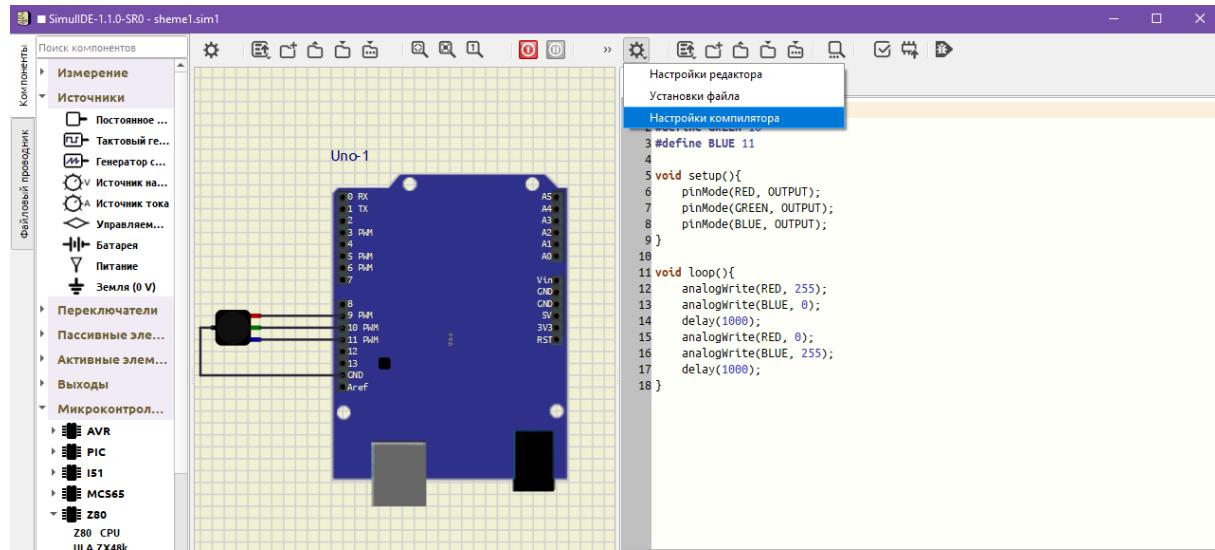
Симулятор электрических схем SimulIDE

Скачать SimulIDE версия 1.1.0: <https://simulide.com/p/downloads/>.

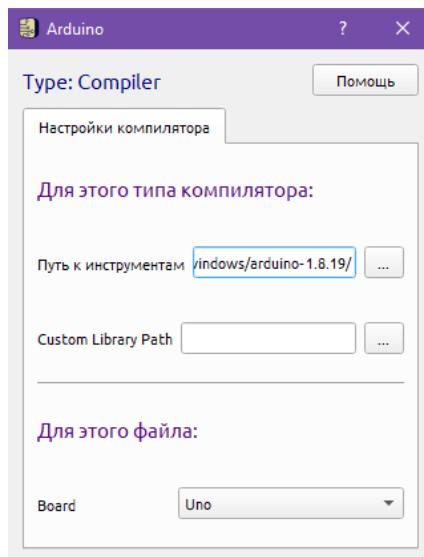
В SimulIDE для работы с платами Arduino необходимо указать путь к ArduinoIDE. Для этого нужно скачать ArduinoIDE версии ниже 2.0. Например, версию 1.8.19: <https://www.arduino.cc/en/software>.

Далее в SimulIDE необходимо открыть схему, создать файл и сохранить его,

а затем открыть настройки компилятора.



В настройках компилятора указать путь к распакованному архиву Arduino IDE.

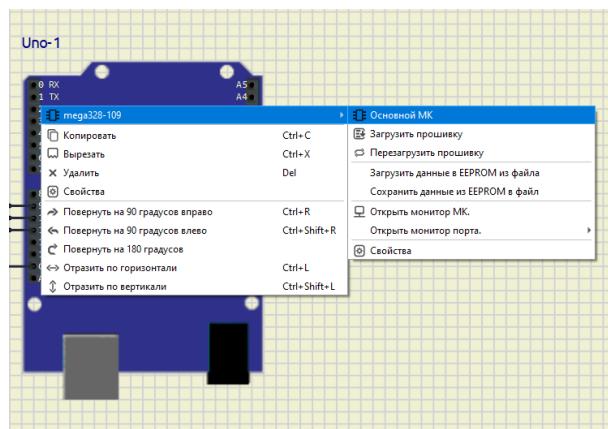


В случае возникновения сложностей обратитесь к документации: <https://simulide.com/p/simulidekb/>.

Примечания

Для того чтобы писать прошивку для двух микроконтроллеров необходимо, чтобы *.ino файлы были в отдельных папках.

Чтобы загрузить прошивку на конкретный микроконтроллер, нужно сначала кликнуть по нему правой кнопкой мыши, а затем выбрать его как основной.



Критерии оценивания

| № п/п | Критерий | Пояснение | Балл |
|--------------|---|--|-----------|
| 1 | Джойстик работает согласно условию | | 10 |
| 2 | Кнопка работает согласно условию | Для каждого нажатия фиксированное действие | 10 |
| 3 | Данные выводятся на дисплей | Соответствует условию — 10 баллов . Частично соответствует — 5 баллов . Не соответствует — 0 баллов | 10 |
| 4 | Расстояние определено корректно | Расстояние выведено на дисплей к установленной погрешностью: +5 баллов . Глубина определена с верной погрешностью: +5 баллов . Расстояние и глубина определены, но погрешность измерений от 2 до 4 см: +3 балла | 10 |
| 5 | RGB-светодиод работает согласно условию | Соответствует условию — 10 баллов . Частично соответствует — 5 баллов . Не соответствует — 0 баллов | 10 |
| ИТОГО | | | 50 |

Решение

Решением задачи является архив с кодом для двух устройств.

Ссылка на вариант решения от разработчиков с комментариями: <https://disk.yandex.ru/d/LH6Mm0qw73y30g>.

Задача 3.2.1.4. Электроника (50 баллов)

Темы: схемотехника, принципиальная схема, печатные платы.

Условие

Участникам предложен проект платы модуля управления моторами для подводного аппарата. К сожалению, принципиальная схема устройства была утеряна.

Задача: по предоставленному проекту печатной платы и слоев дорожек разработать принципиальную схему устройства.

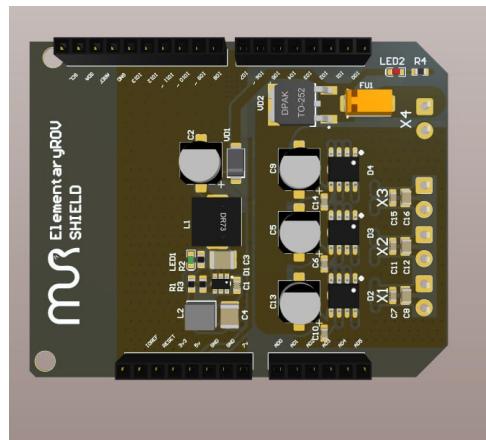


Рис. 3.2.14. Общий вид платы

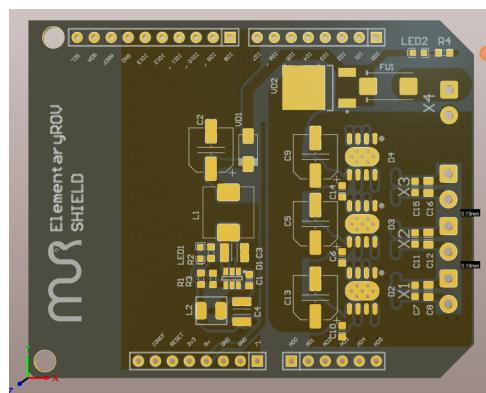


Рис. 3.2.15

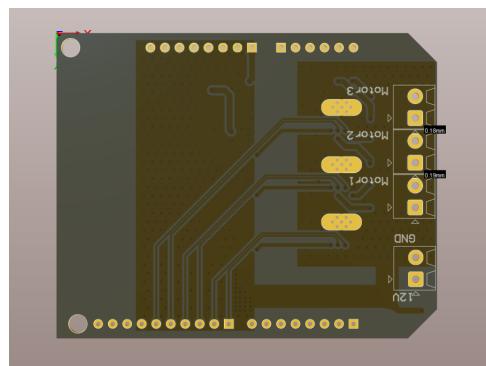


Рис. 3.2.16

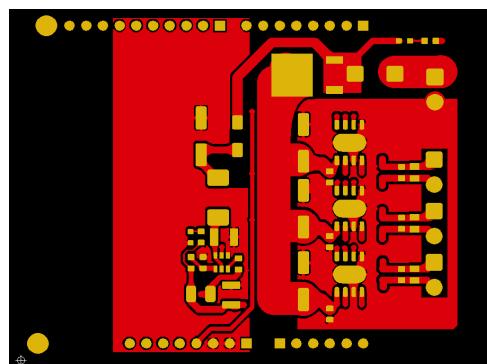


Рис. 3.2.17. Нижний слой платы

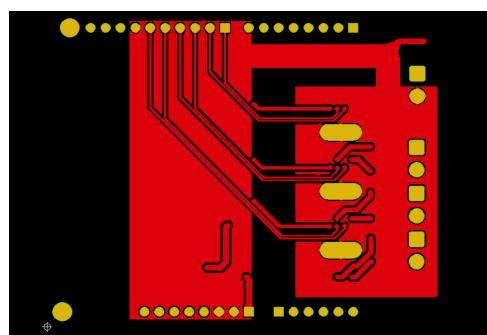


Рис. 3.2.18. Верхний слой платы

Таблица 3.2.4. Список компонентов

| Comment | Value | Designator | Footprint | Quantity |
|------------------------|-----------|-----------------|----------------|----------|
| 100n, X7R, 50V | 100n | C1 | C0603 | 1 |
| EEEFK1E101SP | 100 μ | C2, C5, C9, C13 | SMD, 6.3×5.8mm | 4 |
| 22 μ F, X5R, 25V | 22 μ | C3 | C1210 | 1 |
| 47uF, X5R, 16V | 47u | C4 | C1210 | 1 |
| 220n, X7R, 50V | 220n | C6, C10, C14 | C0603 | 3 |
| 10n, X7R, 50V | 10n | C7, C11, C15 | C0603 | 3 |
| 470nF, X7R, 16V | 470n | C8, C12, C16 | C0805 | 3 |
| TPS561201DDCT | | D1 | SOT23-6 | 1 |
| A4950E | | D2, D3, D4 | SOIC-8 | 1 |
| Littlefuse 0451010.MRL | | FU1 | CAPMP7343X310N | 1 |
| SLTF032T-101MR45 | 100u | L1 | SMD, 7×7×3.2mm | 1 |
| MCW-0420-4R7-N1-M | 4u7 | L2 | 0420 | 1 |
| Green | | LED1 | LED0603 | 1 |
| Red | | LED2 | LED0603 | 1 |
| 0603_1% | 80K6 | R1 | R0603 | 1 |
| 0603_1% | 1K | R2 | R0603 | 1 |

| Comment | Value | Designator | Footprint | Quantity |
|----------------|-------|------------|----------------|----------|
| 0603_1% | 10K | R3, R4 | R0603 | 2 |
| SS56 | | VD1 | DO-214AC | 1 |
| MBRD2020COT-13 | | VD2 | TO-252-2(DPAK) | 1 |

Требования к заданию

1. В принципиальную схему запрещено добавлять компоненты, не представленные в ней, следует подписать обозначения компонентов и их номиналы согласно правилам.
2. Оформлять принципиальную схему необходимо согласно ГОСТ 2.702-75. ЕСКД. Правила выполнения электрических схем: http://www.robot.bstu.ru/files/GOST/gost_2.702-75.pdf

Критерии оценивания

Критерии оценки задания приведены в файле: <https://disk.yandex.ru/i/MECeVsLohjBxQ>.

Решение

Решением задачи является комплект документации:

- принципиальная схема проекта;
- перечень используемых элементов с указанием их номинала (оформляется второй страницей);
- принципиальная схема оформляется в виде pdf-файла.

В качестве возможного варианта решения приводится работа команды, набравшая максимальное количество баллов (50 баллов): <https://disk.yandex.ru/i/12uLURLE2ahilw>.

3.2.2. Командные задачи

Командные задачи второго этапа инженерного тура открыты для решения. Соревнование доступно на платформе Яндекс.Контест: <https://contest.yandex.ru/contest/69914/enter/>.

Задача 3.2.2.1. Техническое задание (100 баллов)

Темы: разработка конструкторской документации, ТЗ, ТНПА.

Условие

1. Важным навыком для инженеров является разработка технической документации. Если изделие изготавливается для себя, то можно ограничиться эс-

кизом или чертежом, но когда в процесс проектирования включено хотя бы несколько человек, когда есть заказчик или когда разрабатываемое устройство состоит из большого числа деталей и подсистем, тогда без полного комплекта документации не обойтись.

2. Одним из важнейших документов на начальной стадии разработки является техническое задание (ТЗ). ТЗ обычно разрабатывается инженерами совместно с заказчиком. И именно по ТЗ заказчик потом принимает опытный образец или готовый продукт.
3. В техническом задании отражены все основные требования к устройству, поэтому, чтобы написать ТЗ, необходимы усилия всех разработчиков (электронщики, конструкторы, программисты и др.).
4. В Российской Федерации создана система стандартизации — ГОСТы, которым должна соответствовать вся конструкторская и технологическая документация. Это позволяет избежать разнотечений в процессе разработки ТЗ. Стандарты сделаны прежде всего для того, чтобы облегчить жизнь разработчикам и всем вовлеченным в процесс проектирования и производства специалистам.
5. В рамках командной задачи участникам предлагается разработать ТЗ к существующему управляемому необитаемому подводному аппарату: <https://www.ecagroup.com/en/solutions/rovingbat-hybrid-rov>. «Заказчик» передает команде картинку с аппаратом и дает краткое описание того, как должен выглядеть и работать робот, а участники должны со слов «заказчика» разработать ТЗ на опытно-конструкторскую работу, результатом которой является опытный образец.
6. В качестве существующего аппарата выбран ROVINGBAT компании ECA Group. Это один из известных подводных роботов в мире, признанный стандарт качества в своей области. Для разработки ТЗ можно использовать любую доступную информацию об аппарате ROVINGBAT.
7. ТЗ будет оцениваться по определенным критериям. Подготовлен специальный лист оценки, который поможет правильно разработать ТЗ. Максимальное количество баллов за данное задание по критериям — 80. Оценка выполняется вручную экспертами, после чего приведенный результат нормируется на 100 баллов.
8. В листе оценки есть формальные критерии (наличие перечня терминов, наличие перечня сокращений и т. п.), а есть содержательные критерии (характеристики изделия, конструктивные требования и т. п.). Кроме того, указаны пункты ГОСТов, на которые эксперты опираются при проверке работ.
9. ТЗ необходимо загрузить в формате *.doc или *.docx. Размер файла должен быть менее 5 Мб. Все тексты будут проверяться на плагиат. Допустимая норма заимствований 20%.

Критерии оценивания

Ознакомиться с критериями оценки можно по ссылке: <https://disk.yandex.ru/i/DyIFySMiR1hbLA>.

Решение

Решением задачи является документ, составленный по указанным в критериях оценивания требованиям.

В качестве возможного варианта решения приводится работа команды, набравшая максимальное количество баллов (97,5 из 100): <https://disk.yandex.ru/i/zaqiNLpo0KFZEw>.

4. Заключительный этап

4.1. Работа наставника НТО при подготовке к этапу

На этапе подготовки к заключительному этапу НТО наставник решает две важные задачи: помочь 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/72663/enter/>.

4.2.1. Информатика. 8–11 классы

Задача 4.2.1.1. Обгон катера (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

Два катера начинают движение параллельными курсами на расстоянии r м друг от друга. Правому катеру требуется совершить маневр с поворотом на 90° влево, тем самым он пересечет траекторию левого катера. Правый катер движется быстрее, поэтому он может опередить левый катер и далее выполнить поворот по дуге окружности. Чтобы маневр был безопасным, требуется, чтобы в момент пересечения траекторий катеров расстояние между ними также было не меньше, чем r .

Напишите программу, которая найдет, через какое время после начала движения правый катер должен начать поворот.

Для лучшего понимания посмотрите схему, представленную на рис. 4.2.1.



Рис. 4.2.1

Левый катер проплыл по прямой от точки A к точке O со скоростью u м/с. Правый катер за это же время проплыл по прямой со скоростью v м/с от точки B .

до точки C и далее по дуге окружности от точки C до точки D . Программа должна найти время, за которое правый катер проплынет отрезок BC .

Подсказка. Чтобы использовать в программе на Python константу π , напишите в первой строчке следующий фрагмент кода.

```
1 from math import pi
```

Формат входных данных

В одной строке через пробел на вход поступает три натуральных числа r, u, v — расстояние между катерами и радиус поворота, скорость левого катера, скорость правого катера, $100 \leq r \leq 10\,000$, $1 \leq u < v \leq 20$.

Формат выходных данных

Выведите одно вещественное число — время, за которое правый катер проплынет отрезок BC . Ответ будет засчитан, если разность между выведенным числом и точным значением не превысит 0,001.

Примеры

Пример №1

| Стандартный ввод |
|--------------------|
| 200 4 5 |
| Стандартный вывод |
| 251.32741228718345 |

Примечания

Поясним ответ к тестовому примеру.

Правый катер проплыл $251,327\,412\,287\,183\,45$ с по прямой и

$$\frac{200\pi}{2 \cdot 5} = 62,831\,853\,071\,795\,86 \text{ с по дуге.}$$

Тогда левый катер плыл по прямой

$$251,327\,412\,287\,183\,45 + 62,831\,853\,071\,795\,86 = 314,159\,265\,358\,979\,3 \text{ с.}$$

Тогда длина отрезка AO равна

$$4 \cdot 314,159\,265\,358\,979\,3 = 1\,256,637\,061\,435\,917\,3,$$

а отрезка BC

$$251,327\,412\,287\,183\,45 \cdot 5 = 1\,256,637\,061\,435\,917\,3.$$

Длины отрезков совпали, то есть время соответствует траекториям на схеме.

Решение

Обозначим за x длину отрезка BC . Дуга составляет четверть окружности, и ее длина равна $\pi \frac{r}{v}$. Таким образом, расстояние, пройденное правым катером, равно

$$x + \pi \frac{r}{2}.$$

Время движения правого катера равно

$$\frac{x + \pi \frac{r}{2}}{v} = \frac{2x + \pi r}{2v}.$$

Левый катер за это время проплыл расстояние $\frac{u(2x + \pi r)}{2v}$, которое равно x , что позволяет записать следующее уравнение.

$$x = \frac{u(2x + \pi r)}{2v}.$$

Решив полученное уравнение, получим формулу для вычисления x .

$$x = \frac{\pi r u}{2(v - u)}.$$

Чтобы найти искомое время, x надо поделить на v .

Пример программы-решения

Ниже представлено решение на языке Python.

Python

```

1  from math import pi
2  r, u, v = map(int, input().split())
3  ans = (pi*r*u) / (2*(v-u)*v)
4  print(ans)

```

Критерии оценивания

Первый тест совпадает с тестом из условия задачи. Баллы за него не начисляются.

Программа проверяется на 10 тестах. Успешное прохождение каждого теста оценивается в 1 балл.

Задача 4.2.1.2. Выбор режима движения (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

Гоночный автомобиль должен проехать по ровной трассе s км. Это расстояние требуется проехать как можно скорее, однако расход топлива на километр обычно возрастает с увеличением скорости, и имеющегося запаса топлива может не хватить, чтобы проехать всю трассу на максимальной скорости. У автомобиля n режимов движения. В i -том режиме автомобиль тратит r_i мл топлива, чтобы проехать 1 км за t_i с. Бак автомобиля вмещает v л топлива.

Напишите программу, которая найдет, за какое минимальное время автомобиль сможет преодолеть требуемую дистанцию. Гарантируется, что емкости бака хватит, чтобы проехать s км хотя бы в самом экономичном режиме. Расстояние, проезжаемое автомобилем в каждом из режимов, может не быть целым.

Подсказка. Эта задача относится к классу задач линейного программирования. Неизвестными переменными x_1, \dots, x_n здесь будут расстояния, пройденные автомобилем в каждом из режимов. В задаче два ограничения: на расстояние и на объем бака. В теории линейного программирования доказывается, что количество ненулевых значений переменных в оптимальном ответе будет меньше или равно, чем количество ограничений. Таким образом, для достижения оптимального времени в этой задаче автомобиль может двигаться не более, чем в двух различных режимах.

Формат входных данных

В первой строке на вход поступают три натуральных числа n , s , v — количество режимов движения, длина дороги и имеющееся количество топлива, $2 \leq n \leq 100$, $1 \leq s \leq 1000$, $1 \leq v \leq 1\,000\,000$.

Далее в n строках записаны описания режимов движения. Описание i -го режима состоит из двух чисел t_i и r_i — время, за которое автомобиль проезжает 1 км в этом режиме, и объем сжигаемого за это время топлива, $10 \leq t_i \leq 200$, $10 \leq r_i \leq 1000$.

Гарантируется, что $s \cdot \min r_i \leq v$, то есть требуемое расстояние можно проехать в самом экономичном режиме.

Формат выходных данных

Выведите одно вещественное число — наименьшее возможное время, за которое автомобиль преодолеет заданное расстояние. Ответ будет засчитан, если разность между выведенным числом и точным значением не превысит 0,001.

Примеры

Пример №1

Стандартный ввод

| |
|-------------|
| 3 100 15000 |
| 60 150 |
| 50 200 |
| 100 50 |

| |
|--------------------------|
| Стандартный вывод |
| 6000 |

Пример №2

| |
|--|
| Стандартный ввод |
| 7 100 15000 20 300 69 100 30 250 60 150 50 200 100 50 80 75 |

| |
|--------------------------|
| Стандартный вывод |
| 5599.99999999999 |

Примечания

Рассмотрим первый тестовый пример. Чтобы получить оптимальное время, можно использовать второй и третий режим движения. Во втором режиме автомобиль будет ехать $100 \cdot \frac{2}{3}$ км, а в третьем — $100 \cdot \frac{1}{3}$ км. При этом будет потрачено ровно 15 000 мл топлива.

$$100 \cdot \frac{2}{3} \cdot 100 + 100 \cdot \frac{1}{3} \cdot 250 = 45000 \cdot \frac{1}{3} = 15000.$$

Время движения составит:

$$100 \cdot \frac{2}{3} \cdot 69 + 100 \cdot \frac{1}{3} \cdot 30 = 5600.$$

Выведенный ответ отличается от точного значения менее, чем на 0,001, поэтому он считается верным.

Можно показать, что меньшее значения времени получить невозможно.

Во втором тестовом примере оптимальным вариантом будет проехать весь путь в первом режиме движения.

Решение

С учетом ограничений решение этой задачи может заключаться в переборе всех пар режимов движения. Обозначим объем сжигаемого топлива и время для первого режима за r_i и t_i , а для второго режима — за r_j и t_j . Будем считать, что $sr_i < v$, а $sr_j > v$. Если $sr_j \geq v$, то достаточно использовать один режим движения.

Обозначим за x расстояние, пройденное в первом режиме. Тогда расстояние, пройденное во втором режиме, будет равно $s - x$. Тогда $xr_i + (s - x)r_j = v$. Решив

уравнение, получим формулу

$$x = \frac{v - sr_i}{r_j - r_i}.$$

Из нее можно получить время движения $t = xt_i + (s - x)t_j$. Требуется перебрать все пары режимов и найти минимум из полученных значений t . Необходимо также учесть, что может оказаться оптимальным проехать все расстояние, используя один режим.

Пример программы-решения

Ниже представлено решение на языке Python.

Python

```

1  n, s, v = map(int, input().split())
2  p = [tuple(map(int, input().split())) for _ in range(n)]
3  p.sort(key = lambda x:x[1])
4  ans = 10**18
5  for i in range(n):
6      if p[i][1]*s <= v and p[i][0]*s < ans:
7          ans = p[i][0]*s
8  for i in range(n):
9      if p[i][1]*s>=v:
10         break
11  for j in range(i+1, n):
12      if p[j][1]*s>v:
13          x = (v-p[i][1]*s) / (p[j][1]-p[i][1])
14          t = p[j][0]*x+p[i][0]*(s-x)
15          if t<ans:
16              ans = t
17  print(ans)

```

Критерии оценивания

Два первых теста совпадают с тестами из условия задачи. Баллы за них не начисляются.

Программа проверяется на 15 тестах. Успешное прохождение каждого теста оценивается в 1 балл.

В тестах №№ 3–8 задано ровно два режима движения.

Задача 4.2.1.3. Ловушки (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

Петя опять прогулял информатику, и Антон в учебно-воспитательных целях создал для него виртуальный мир k -значных ловушек. Мир выглядит как последовательность комнат, в каждой из которых находится k -значная ловушка. У каждой ловушки имеется k состояний, пронумерованных цифрами от 0 до $k-1$. Петя должен пройти по всем комнатам от первой до последней. Если в некоторой комнате имеется ловушка с состоянием, равным нулю, то Петя, войдя в эту комнату, проваливается в ловушку и переносится в начало последовательности, после чего начинает двигаться по комнатам заново. Состояние ловушки при этом изменяется на $k-1$.

Если же ловушка в некоторой комнате имела состояние $s > 0$, то Петя проходит через эту комнату, однако состояние ловушки при этом становится равным $s-1$.

Петя может досрочно покинуть этот странный мир, если правильно ответит, сколько раз ему придется попадать в ловушки, перед тем как он сможет дойти до выхода. Он уже осознал ошибочность своего поведения и теперь просит написать программу, которая по текущему состоянию ловушек в комнатах даст ответ на поставленный вопрос. Программа должна предоставить решение для t независимых тестовых случаев.

Формат входных данных

В первой строке на вход подается одно натуральное число t — количество тестовых случаев, $1 \leq t \leq 100$. Далее записаны сами тесты, каждый из которых занимает две строки. В первой строке записаны два натуральных числа n и k — количество комнат и показатель мира k -значных ловушек, $1 \leq n \leq 500$, $2 \leq k \leq 10$.

Во второй строке записаны n целых чисел a_1, \dots, a_n — состояния ловушек в комнатах в некоторый момент времени, когда Петя стоит у входа в первую комнату, $0 \leq a_i < k$.

Гарантируется, что числа подобраны таким образом, что ответ не превысит 10^{18} .

Формат выходных данных

Для каждого тестового случая программа должна вывести в отдельной строке одно целое число, равное количеству попаданий в ловушки до момента выхода из последней комнаты.

Примеры

Пример №1

| Стандартный ввод |
|---|
| 3 4 3 2 1 0 1 3 2 1 0 0 4 9 1 1 1 1 |

Стандартный вывод

```
3
6
0
```

Примечания

Рассмотрим первый тестовый пример. В свой первый проход Петя пройдет две комнаты и попадет в ловушку в третьей.

Состояние ловушек при этом станет $1 \ 0 \ 2 \ 1$. В свой второй проход Петя пройдет одну комнату и попадет в ловушку во второй.

Состояние ловушек при этом станет $0 \ 2 \ 2 \ 1$. В свой третий проход Петя попадет в ловушку в самой первой комнате.

Состояние ловушек при этом станет $2 \ 2 \ 2 \ 1$. Теперь открытых ловушек нет, и Петя дойдет до выхода. В ответ выводится количество попаданий в ловушки, то есть 3.

Решение

Задача может быть решена с использованием стандартного алгоритма целочисленного бинарного поиска. Функция `test(s)` будет возвращать истину, если, входя в первую комнату s раз, Петя сможет выйти из последней комнаты. Эта функция, очевидно, является монотонной, поэтому применение бинарного поиска является корректным. Ответом будет максимальное значение s , при котором `test(s)` вернет ложь.

Сама функция `test` может быть реализована циклически. Если Петя войдет в некоторую комнату s раз, то он попадет в ловушку $\left\lceil \frac{s-a}{k} \right\rceil$ раз, где a — начальное состояние ловушки, а k — ее параметр. Тогда в следующую комнату Петя войдет $s - \left\lceil \frac{s-a}{k} \right\rceil$ раз и так далее. Если переменная s станет равна нулю, то Петя не сможет дойти до выхода из последней комнаты.

Пример программы-решения

Ниже представлено решение на языке Python.

Python

```
1 def test(s):
2     for a in lst:
3         s -= (s-a+k-1)//k
4     return s>0
5 t = int(input())
6 for _ in range(t):
7     n, k = map(int, input().split())
8     lst = [int(x) for x in input().split()]
9     left = 0
10    right = 10**18
```

```

11  while right-left > 1:
12      mid = (left+right)//2
13      if test(mid):
14          right=mid
15      else:
16          left=mid
17  print(left)

```

Критерии оценивания

Первый тест совпадает с тестом из условия задачи. Баллы за него не начисляются.

Программа проверяется на 20 тестах. Успешное прохождение каждого теста оценивается в 1 балл.

В тестах №№ 2–9 во всех тестовых случаях $k = 2$. Дополнительно в тестах №№ 2–5 количество комнат в каждом тестовом случае не превосходит 10.

В тестах №№ 10–15 количество комнат в каждом тестовом случае не превосходит 15.

Задача 4.2.1.4. Терракотовая армия (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: 128 Мбайт.

Условие

Антон разрабатывает виртуальный музей с самыми известными культурными достопримечательностями человечества. Сегодня его внимание привлекла терракотовая армия — тысячи глиняных статуй воинов, расположенных у мавзолея Цинь Шихуанди — основателя империи Цинь.

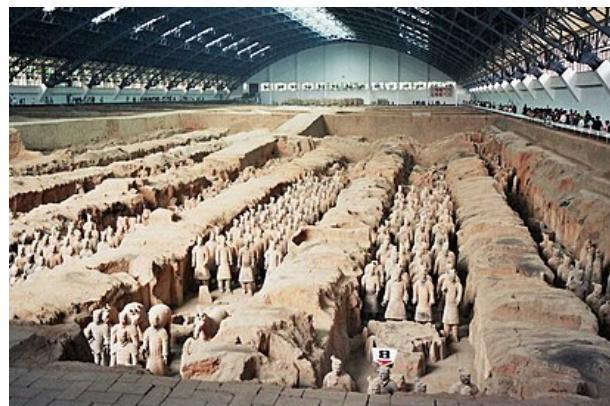


Рис. 4.2.2

Чтобы сделать экспозицию более динамичной, он решил, что воины должны двигаться по следующей схеме. Представим себе поле, состоящее из n ячеек, расположенных в одну линию. В начальный момент времени в каждой ячейке поля находится один воин. Далее за каждый такт происходят следующие изменения.

Воин, стоящий на первой клетке падает, сраженный, и пропадает с поля, а оставшиеся воины перемещаются так, чтобы выполнялись следующие условия:

- За один шаг воин перемещается в следующую клетку. За такт каждый воин может сделать несколько шагов или ни одного.
- В первой и последней клетках поля всегда должно находиться по одному воину.
- Количество пустых клеток между любой парой соседних воинов должно быть примерно одинаковым. Это означает, что наибольшее количество свободных клеток между двумя соседними воинами и наименьшее должны различаться не более, чем на единицу.
- Суммарное количество шагов, сделанных всеми воинами, на каждом отдельном такте должно быть минимальным.

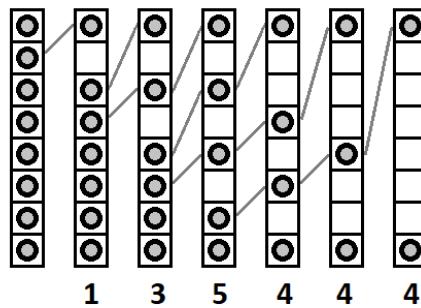


Рис. 4.2.3

Для лучшего понимания посмотрите на схему перемещений за 6 тактов при $n = 8$ (см. рис. 4.2.3). Числа на схеме указывают суммарное количество шагов.

Антона заинтересовал вопрос, а сколько всего шагов сделают воины на каждом такте от 1 до $n - 2$. Напишите программу, которая даст на него ответ.

Формат входных данных

На вход поступает единственное натуральное число n , $3 \leq n \leq 200\,000$.

Формат выходных данных

Программа должна вывести в одной строке через пробел $n - 2$ натуральных числа — суммарное количество шагов, сделанное воинами на каждом такте демонстрации от 1 до $n - 2$.

Примеры

Пример №1

| Стандартный ввод |
|-------------------|
| 8 |
| Стандартный вывод |
| 1 3 5 4 4 4 |

Примечания

Тестовый пример рассмотрен на схеме 4.2.3.

Решение

Для решения этой задачи можно применить следующий подход. Пронумеруем все позиции на поле, начиная с нуля. Будем считать сумму номеров позиций, на которых стоят воины после каждого такта. При этом будем учитывать снятых с поля воинов как стоящих на позиции $n-1$. Тогда количество шагов на каждом такте будет разностью сумм номеров между текущим состоянием и предыдущим.

Пусть i — номер текущего такта. Воина, стоящего в клетке 0, не учитываем в сумме. Тогда количество воинов будет равно $n-1-i$. Представим число $n-1$ в виде

$$n-1 = m(n-1-i) + k.$$

Это означает, что $n-1-i-k$ воинов находятся на расстоянии m друг от друга, а k воинов — на расстоянии $m+1$. Чтобы получить такую расстановку, можно сначала расставить $n-1-i$ воинов на позиции $m, 2m, 3m$ и так далее, а после дополнительно сдвинуть k воинов на 1, 2, 3 и так далее. Тогда сумма номеров позиций может быть вычислена с использованием формулы суммы арифметической прогрессии следующим образом:

$$i(n-1) + m(n-i)(n-i-1)/2 + k(k+1)/2.$$

Пример программы-решения

Ниже представлено решение на языке Python.

Python

```

1 n = int(input())
2 sm=[n*(n-1)//2]
3 for i in range(1,n-1):
4     m = (n-1) // (n-1-i)
5     k = (n-1) % (n-1-i)
6     sm.append(i*(n-1) + m*(n-i)*(n-i-1)//2 + k*(k+1)//2)
7 print(' '.join(str(sm[i+1]-sm[i]) for i in range(n-2)))

```

Критерии оценивания

Первый тест совпадают с тестом из условия задачи. Баллы за него не начисляются.

Программа проверяется на 25 тестах. Успешное прохождение каждого теста оценивается в 1 балл.

В тестах №№ 2–11 размер поля n не превосходит 100.

В тестах №№ 12–16 размер поля n не превосходит 10 000.

Задача 4.2.1.5. Усталый Иван-царевич (30 баллов)

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

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

Ограничение по времени выполнения программы: 2 с.

Ограничение по памяти: 128 Мбайт.

Условие

«Хорошо, хоть яйца не разбегаются», — устало подумал Иван-царевич, отстреливая стотысячного зайца.

Много тысячелетий назад Кощей Бессметный осознал свою уязвимость. Герою достаточно пронзить стрелой зайца, утку, разбить яйцо — и все. Тогда он решил усилить свою защиту. Он создал много объектов трех видов: «яйца», «зайцы», «утки». При этом каждый объект может содержать в себе произвольное количество других объектов. Например, в зайце могут скрываться еще два зайца, утка и яйцо, внутри них — новые зайцы утки и яйца и так далее. После поражения внешнего объекта все внутренние немедленно высекают наружу. При этом зайцы и утки быстро разбегаются или разлетаются в разные стороны, и попробуй их догони! Яйца же остаются на месте, и бегать за ними не придется. Единственным самым внешним объектом является некоторое яйцо, а смерть Кощея наступит, когда будут поражены все объекты.

Конечно же, это была завязка сюжета для логической игры с элементами виртуальной реальности.

В нашей игре объединенный штаб всех сил добра разрабатывает план сражения с Кощеем. Команде магических хакеров удалось выкрасть схему размещения объектов в защите Кощея. При ближайшем рассмотрении рунические символы на ней оказались обычновенными скобками. Яйцо записывается как пара из круглых скобок `()`, заяц — квадратных `[]`, а утка — фигурных `{ }`. Вложенность скобок показывает вложенность объектов. Например, последовательность `([{ }] [] { ([{ }]) })` задает яйцо, в котором находится первый заяц с двумя утками внутри, второй заяц без вложенных объектов и утка, внутри которой находится яйцо, в том яйце еще один заяц, а в нем — еще одна утка.

Команда магов-артефакторов создала волшебные стрелы, которые никогда не промахиваются, а команда магов-лекарей усилила способности и силу Ивана-царевича. Теперь сражение можно представить в виде следующей модели. В начале

каждой минуты Иван-царевич произвольно выбирает и поражает один из объектов, находящихся на поверхности. Далее в течении минуты он выбирает и догоняет новый объект, чтобы поразить его в начале следующей минуты. Усиленные способности Ивана-царевича позволяют ему всегда догнать зайца или утку за минуту, независимо от расстояния на которое те убежали, однако на это потребуется некоторое количество энергии. Будем считать, что если заяц или утка оказались на поверхности в момент времени i , а поражены были в момент времени j , то это потребовало $j - i$ единиц энергии. Яйца не убегают, поэтому их поиск не требует энергии.

Вы — маг-стратег, от которого требуется составить план сражения, то есть последовательность поражения объектов, при которой Иван-царевич потратит минимальное количество энергии. Поскольку объектов может быть много, вам придется написать программу для решения указанной задачи.

Формат входных данных

На вход поступает правильная скобочная последовательность с тремя видами скобок. Длина последовательности не превосходит 300 000. Первый и последний символ — это круглая открывающая и соответствующая ей закрывающая скобка. Количество уровней вложенности скобок не превосходит 1 000.

Формат выходных данных

Выведите одно целое число — минимальное количество энергии, которое потратит Иван-царевич для поражения всех объектов.

Примеры

Пример №1

| |
|--------------------------|
| Стандартный ввод |
| ([{}{}][[]{{([{}])}}) |
| Стандартный вывод |
| 11 |

Примечания

Рассмотрим тестовый пример ([{}{}][[]{{([{}])}}).

В момент времени 0 Иван-царевич разбьет внешнее яйцо. На поверхности окажутся три объекта: [{}{}], [] и {{([{}])}}.

В момент времени 1 Иван-царевич может поразить утку и потратить на это $1 - 0 = 1$ единицу энергии. На поверхности останутся объекты: [{}{}], [] и {{([{}])}}.

В момент времени 2 Иван-царевич может поразить второго зайца и потратить на это $2 - 0 = 2$ единицы энергии. На поверхности останутся объекты: [{}{}] и {{([{}])}}.

В момент времени 3 Иван-царевич может поразить оставшегося зайца и потратить на это $3 - 0 = 3$ единицы энергии. На поверхности останутся объекты: $\{ \}$, $\{ \}$ и $([\{ \}])$. Заметим, что утки появились в момент времени 3.

В момент времени 4 Иван-царевич может поразить одну из уток и потратить на это $4 - 3 = 1$ единицу энергии. На поверхности останутся объекты: $\{ \}$ и $([\{ \}])$.

В момент времени 5 Иван-царевич может поразить оставшуюся утку и потратить на это $5 - 3 = 2$ единицы энергии. На поверхности останется одно яйцо: $([\{ \}])$.

В момент времени 6 Иван-царевич разобьет яйцо и не потратит на это энергии. На поверхности останется один заяц: $\{ \}$.

В момент времени 7 Иван-царевич поразит зайца и потратит на это $7 - 6 = 1$ единицу энергии. На поверхности останется одна утка: $\{ \}$.

В момент времени 8 Иван-царевич поразит последнюю утку и потратит на это $8 - 7 = 1$ единицу энергии.

Всего Иван-царевич потратил $1 + 2 + 3 + 1 + 2 + 1 + 1 = 11$ единиц энергии.

Решение

Сделаем несколько замечаний. Во-первых, каждое яйцо может рассматриваться как независимый объект, поскольку его уничтожение можно отложить на произвольное время без потерь энергии.

Во-вторых, в каждый момент времени на поверхности должно находиться как можно меньше объектов, поэтому, начав ликвидацию некоторого объекта, следует немедленно уничтожать и все вложенные объекты, кроме яиц.

В-третьих, следует устранивать объекты в порядке возрастания их размера с учетом вложенности.

Для решения задачи можно использовать алгоритм обхода дерева в глубину. Само дерево здесь неявно задано с использованием скобочного выражения, поэтому текущей позицией вершины в дереве будем считать позицию некоторой открывающей скобки. В прилагаемой программе эта позиция хранится в переменной `ср`. Каждый рекурсивный вызов будет обрабатывать некоторое правильное скобочное подвыражение, изменяя переменную `ср`.

В переменной `ans` накапливается ответ. Функция `dfs` возвращает размер поддерева, при этом учитываются только фигурные и квадратные скобки в соответствии с первым замечанием.

Время нахождения на поверхности непосредственных потомков текущего объекта отмеряется от момента уничтожения этого объекта в соответствии со вторым замечанием.

Для подсчета времени нахождения на поверхности непосредственных потомков упорядочим их по возрастанию количества вложенных объектов. Если обозначить эти значения c_1, c_2, \dots, c_m , то первый объект будет уничтожен через 1 с, второй — через $(1 + c_1)$ с, третий — через $(1 + c_1 + c_2)$ с и так далее. Эти величины добавляются к ответу.

Пример программы-решения

Ниже представлено решение на языке Python.

Python

```

1  seq = input()
2  cp = 0
3  ans = 0
4  def dfs():
5      global cp,ans
6      cp+=1
7      cnt = []
8      while True:
9          if seq[cp]==')':
10              dfs()
11          elif seq[cp]=='[' or seq[cp]=='{':
12              cnt.append(dfs())
13          else:
14              break
15      cnt.sort()
16      s = 1
17      for c in cnt:
18          ans+=s
19          s+=c
20      cp+=1
21      return s
22  dfs()
23  print(ans)

```

Критерии оценивания

Первый тест совпадают с тестом из условия задачи. Баллы за него не начисляются.

Программа проверяется на 40 тестах. Успешное прохождение каждого теста оценивается в 0,5 балла. Если программа прошла все тесты, то начисляется еще 10 баллов.

В тестах №№ 2–11 длина строки не превосходит 300.

В тестах №№ 12–21 длина строки не превосходит 3 000.

В тестах №№ 22–31 длина строки не превосходит 30 000.

4.2.2. Физика. 8–9 классы

Задача 4.2.2.1. Шарик и кубик (15 баллов)

Тема: плотность.

Условие

Сосуд доверху заполнили жидкостью плотности $\rho = 1,2 \text{ г/см}^3$. Далее в него поместили шарик, объем которого в четыре раза меньше объема сосуда. Средняя плотность содержимого сосуда стала равна $\rho_1 = 1,5 \text{ г/см}^3$. Определите плотность шарика $\rho_{\text{ш}}$. Если вместо шарика в сосуд поместить кубик такого же объема, то средняя плотность содержимого сосуда станет равна $\rho_2 = 1,7 \text{ г/см}^3$. Определите плотность кубика $\rho_{\text{к}}$. Какой станет средняя плотность ρ_3 содержимого сосуда, если в него поместить одновременно шарик и кубик?

Решение

Обозначим объем шарика и кубика за V , тогда объем сосуда будет $4V$. Для средней плотности содержимого сосуда и шарика запишем

$$\rho_1 = \frac{\rho \cdot 3V + \rho_{\text{ш}}V}{4V} = \frac{3\rho + \rho_{\text{ш}}}{4}.$$

Плотность шарика:

$$\rho_{\text{ш}} = 4\rho_1 - 3\rho = 2,4 \text{ г/см}^3.$$

Для средней плотности содержимого сосуда и кубика запишем

$$\rho_2 = \frac{\rho \cdot 3V + \rho_{\text{к}}V}{4V} = \frac{3\rho + \rho_{\text{к}}}{4}.$$

Плотность кубика:

$$\rho_{\text{к}} = 4\rho_2 - 3\rho = 3,2 \text{ г/см}^3.$$

Для средней плотности содержимого сосуда, шарика и кубика запишем

$$\rho_3 = \frac{\rho \cdot 2V + \rho_{\text{ш}}V + \rho_{\text{к}}V}{4V} = \frac{2\rho + \rho_{\text{ш}} + \rho_{\text{к}}}{4}.$$

Складывая два первых уравнения и вычитая третье уравнение, получим

$$\rho_1 + \rho_2 - \rho_3 = \rho \rightarrow \rho_3 = \rho_1 + \rho_2 - \rho,$$

$$\rho_3 = 2,0 \text{ г/см}^3.$$

Ответ: $\rho_{\text{ш}} = 2,4 \text{ г/см}^3$, $\rho_{\text{к}} = 3,2 \text{ г/см}^3$, $\rho_3 = 2,0 \text{ г/см}^3$.

Критерии оценивания

| | |
|---|------------------|
| Обозначим объем шарика и кубика за V , тогда объем сосуда будет $4V$ | 1 балл |
| $\rho_1 = \frac{\rho \cdot 3V + \rho_{ш}V}{4V} = \frac{3\rho + \rho_{ш}}{4}$ | 3 балла |
| $\rho_{ш} = 4\rho_1 - 3\rho = 2,4 \text{ г/см}^3$ | 1 балл |
| $\rho_2 = \frac{\rho \cdot 3V + \rho_{к}V}{4V} = \frac{3\rho + \rho_{к}}{4}$ | 3 балла |
| $\rho_{к} = 4\rho_2 - 3\rho = 3,2 \text{ г/см}^3$ | 1 балл |
| $\rho_3 = \frac{\rho \cdot 2V + \rho_{ш}V + \rho_{к}V}{4V} = \frac{2\rho + \rho_{ш} + \rho_{к}}{4}$ | 3 балла |
| $\rho_1 + \rho_2 - \rho_3 = \rho \rightarrow \rho_3 = \rho_1 + \rho_2 - \rho$ | 2 балла |
| $\rho_3 = 2,0 \text{ г/см}^3$ | 1 балл |
| Всего | 15 баллов |

Задача 4.2.2.2. Нелинейные элементы (15 баллов)

Темы: закон Ома для участка цепи, мощность.

Условие

Две нелинейных элемента имеют вольтамперные характеристики $U_1 = \alpha_1 I^2$ и $U_2 = \alpha_2 I^2$, где $\alpha_1 = 200 \text{ В/А}^2$, $\alpha_2 = 700 \text{ В/А}^2$. Элементы соединили последовательно и подключили к источнику с напряжением $U = 2,25 \text{ В}$.

1. Определите мощности, выделяющиеся на нелинейных элементах.

Затем нелинейный элемент с вольтамперной характеристикой $U_1 = \alpha_1 I^2$, где $\alpha_1 = 200 \text{ В/А}^2$ соединили последовательно с резистором с сопротивлением $R = 90 \text{ Ом}$ и подключили к источнику с напряжением $U = 18 \text{ В}$.

2. Определите ток I в полученной цепи.

3. Определите мощности, выделяющиеся на нелинейном элементе $P_{нэ}$ и резисторе P_R .

Решение

При последовательном соединении элементов в цепи для общего тока и общего напряжения имеем:

$$I = I_1 = I_2,$$

$$U = U_1 + U_2.$$

Ток, текущий через нелинейные элементы:

$$I = \left(\frac{U}{\alpha_1 + \alpha_2} \right)^{1/2},$$

$$I = 50 \text{ мА.}$$

Мощности, выделяющиеся на нелинейных элементах:

$$P = IU = \alpha I^3,$$

$$P_1 = I_1 U_1 = \alpha_1 I^3 = 25 \text{ мВт},$$

$$P_2 = I_2 U_2 = \alpha_2 I^3 = 87,5 \text{ мВт}.$$

При последовательном соединении нелинейного элемента и резистора имеем:

$$U = \alpha_1 I^2 + IR.$$

Решая квадратное уравнение $\alpha_1 I^2 + IR - U = 0$, получим:

$$I = \frac{-R \pm \sqrt{R^2 + 4\alpha_1 U}}{2\alpha_1}.$$

Так как $I > 0$, то $I = 0,15 \text{ А}$. Мощность, выделяющаяся на нелинейном элементе:

$$P_{\text{НЭ}} = IU_1 = \alpha_1 I^3 = 0,675 \text{ Вт}.$$

Мощность, выделяющаяся на резисторе:

$$P_R = I^2 R = 2,025 \text{ Вт}.$$

Ответ:

1. $P_1 = 25 \text{ мВт}$, $P_2 = 87,5 \text{ мВт}$.
2. $I = 0,15 \text{ А}$.
3. $P_{\text{НЭ}} = 0,675 \text{ Вт}$, $P_R = 2,025 \text{ Вт}$.

Критерии оценивания

| | |
|--|------------------|
| $I = I_1 = I_2$ | 1 балл |
| $U = U_1 + U_2$ | 1 балл |
| $I = \left(\frac{U}{\alpha_1 + \alpha_2} \right)^{1/2}$ | 1 балл |
| $I = 50 \text{ мА}$ | 1 балл |
| $P = IU = \alpha I^3$ | 1 балл |
| $P_1 = I_1 U_1 = \alpha_1 I^3 = 25 \text{ мВт}$ | 1 балл |
| $P_2 = I_2 U_2 = \alpha_2 I^3 = 87,5 \text{ мВт}$ | 1 балл |
| $U = \alpha_1 I^2 + IR$ | 2 балла |
| $I = \frac{-R \pm \sqrt{R^2 + 4\alpha_1 U}}{2\alpha_1}$ | 2 балла |
| Так как $I > 0$, то | 1 балл |
| $I = 0,15 \text{ А}$ | 1 балл |
| $P_{\text{НЭ}} = IU_1 = \alpha_1 I^3 = 0,675 \text{ Вт}$ | 1 балл |
| $P_R = I^2 R = 2,025 \text{ Вт}$ | 1 балл |
| Всего | 15 баллов |

Задача 4.2.2.3. Лед, вода и вода (20 баллов)

Тема: тепловой баланс.

Условие

В сосуд поместили лед, имеющий температуру $t_{\text{л}} = -6^{\circ}\text{C}$. Затем в него налили воду при комнатной температуре $t_{\text{в}} = 24^{\circ}\text{C}$. В результате теплообмена в сосуде установилась температура $t_1 = 6^{\circ}\text{C}$. Определите, во сколько раз масса воды больше массы льда. Далее в сосуд налили еще столько же воды при комнатной температуре. Какая температура t_2 установится в сосуде? Удельная теплоемкость льда $c_{\text{л}} = 2\,100 \text{ Дж}/(\text{кг}\cdot^{\circ}\text{C})$, воды — $c_{\text{в}} = 4\,200 \text{ Дж}/(\text{кг}\cdot^{\circ}\text{C})$, удельная теплота плавления льда $\lambda = 340 \text{ кДж}/\text{кг}$. Теплоемкостью сосуда и тепловыми потерями пренебречь.

Решение

Температура плавления льда $t_0 = 0^{\circ}\text{C}$.

Из уравнения теплового баланса для воды и льда:

$$m_{\text{в}}c_{\text{в}}(t_{\text{в}} - t_1) = m_{\text{л}}c_{\text{л}}(t_0 - t_{\text{л}}) + m_{\text{л}}\lambda + m_{\text{л}}c_{\text{в}}(t_1 - t_0)$$

найдем отношение массы воды к массе льда:

$$\frac{m_{\text{в}}}{m_{\text{л}}} = \frac{-c_{\text{л}}t_{\text{л}} + \lambda + c_{\text{в}}t_1}{c_{\text{в}}(t_{\text{в}} - t_1)},$$

$$\frac{m_{\text{в}}}{m_{\text{л}}} = 5.$$

При доливании воды из уравнения теплового баланса:

$$m_{\text{в}}c_{\text{в}}(t_{\text{в}} - t_2) = (m_{\text{в}} + m_{\text{л}})c_{\text{в}}(t_2 - t_1)$$

определим конечную температуру:

$$t_2 = \frac{t_{\text{в}} \frac{m_{\text{в}}}{m_{\text{л}}} + t_1 \left(\frac{m_{\text{в}}}{m_{\text{л}}} + 1 \right)}{2 \frac{m_{\text{в}}}{m_{\text{л}}} + 1},$$

$$t_2 = 14,2^{\circ}\text{C}.$$

Ответ: $\frac{m_{\text{в}}}{m_{\text{л}}} = 5, t_2 = 14,2^{\circ}\text{C}$.

Критерии оценивания

Температура плавления льда $t_0 = 0^{\circ}\text{C}$

2 балла

$$m_{\text{в}}c_{\text{в}}(t_{\text{в}} - t_1) = m_{\text{л}}c_{\text{л}}(t_0 - t_{\text{л}}) + m_{\text{л}}\lambda + m_{\text{л}}c_{\text{в}}(t_1 - t_0)$$

4 балла

$$\frac{m_{\text{в}}}{m_{\text{л}}} = \frac{-c_{\text{л}}t_{\text{л}} + \lambda + c_{\text{в}}t_1}{c_{\text{в}}(t_{\text{в}} - t_1)}$$

3 балла

$$\frac{m_{\text{в}}}{m_{\text{л}}} = 5$$

2 балла

$$m_{\text{в}}c_{\text{в}}(t_{\text{в}} - t_2) = (m_{\text{в}} + m_{\text{л}})c_{\text{в}}(t_2 - t_1)$$

4 балла

$$t_2 = \frac{t_{\text{в}} \frac{m_{\text{в}}}{m_{\text{л}}} + t_1 \left(\frac{m_{\text{в}}}{m_{\text{л}}} + 1 \right)}{2 \frac{m_{\text{в}}}{m_{\text{л}}} + 1}$$

3 балла

| | |
|--------------------------------------|------------------|
| $t_2 = 14,2 \text{ } ^\circ\text{C}$ | 2 балла |
| Всего | 20 баллов |

Задача 4.2.2.4. Катер (25 баллов)

Тема: кинематика равномерного движения.

Условие

Катер из пункта А в пункт В двигался со скоростью 40 м/мин. Далее из пункта В в пункт С он перемещался со скоростью 69 м/мин. А расстояние из пункта С в пункт D он преодолел со скоростью 60 м/мин.

Определите среднюю скорость катера на всем пути, если расстояние между пунктами А и В составляет 25% всего пути, а перемещение из пункта В в пункт С заняло третью часть всего времени движения.

Решение

Из условия задачи имеем:

движение из пункта А в пункт В ($v_1 = 40 \text{ м/мин}$) $t_1 = \frac{S_1}{v_1} = \frac{S}{4v_1}$;

движение из пункта В в пункт С ($v_2 = 69 \text{ м/мин}$) $S_2 = v_2 t_2 = \frac{v_2 t}{3}$;

движение из пункта С в пункт D ($v_3 = 60 \text{ м/мин}$) $S_3 = S - S_1 - S_2 = \frac{3S}{4} - \frac{v_2 t}{3}$;

$$t_3 = \frac{S_3}{v_3} = \frac{3S}{4v_3} - \frac{v_2 t}{3v_3}.$$

Общее время движения автомобиля

$$t = t_1 + t_2 + t_3 = \frac{S}{4v_1} + \frac{t}{3} + \frac{3S}{4v_3} - \frac{v_2 t}{3v_3}.$$

Так как

$$v_{\text{ср}} = \frac{S}{t},$$

то из уравнения

$$1 = \frac{v_{\text{ср}}}{4v_1} + \frac{1}{3} + \frac{3v_{\text{ср}}}{4v_3} - \frac{v_2}{3v_3}$$

получим

$$v_{\text{ср}} = \frac{4v_1(v_2 + 2v_3)}{3(3v_1 + v_3)},$$

$$v_{\text{ср}} = 56 \text{ м/мин.}$$

Ответ: $v_{\text{ср}} = 56 \text{ м/мин.}$

Критерии оценивания

| | |
|---|------------------|
| $t_1 = \frac{S_1}{v_1} = \frac{S}{4v_1}$ | 2 балла |
| $S_2 = v_2 t_2 = \frac{v_2 t}{3}$ | 2 балла |
| $S_3 = S - S_1 - S_2 = \frac{3S}{4} - \frac{v_2 t}{3}$ | 3 балла |
| $t_3 = \frac{S_3}{v_3} = \frac{3S}{4v_3} - \frac{v_2 t}{3v_3}$ | 2 балла |
| $t = t_1 + t_2 + t_3 = \frac{S}{4v_1} + \frac{t}{3} + \frac{3S}{4v_3} - \frac{v_2 t}{3v_3}$ | 3 балла |
| $v_{cp} = \frac{S}{t}$ | 2 балла |
| $1 = \frac{v_{cp}}{4v_1} + \frac{1}{3} + \frac{3v_{cp}}{4v_3} - \frac{v_2}{3v_3}$ | 3 балла |
| $v_{cp} = \frac{4v_1(v_2+2v_3)}{3(3v_1+v_3)}$ | 5 баллов |
| $v_{cp} = 56 \text{ м/мин}$ | 3 балла |
| Всего | 25 баллов |

Задача 4.2.2.5. Все дело в глицерине (25 баллов)

Темы: гидростатика, сила Архимеда.

Условие

В сосуд с вертикальными стенками и площадью основания $S = 300 \text{ см}^2$ налили глицерин плотностью $\rho_g = 1,2 \text{ г/см}^3$. Затем в него опустили плотик ($\rho_p = 0,4 \text{ г/см}^3$). В результате уровень глицерина в сосуде изменился на $\Delta h = 1 \text{ см}$. Определите объем плотика V_p . Затем на плотик сверху положили кусок льда ($\rho_l = 0,9 \text{ г/см}^3$). В результате плотик погрузился в глицерин полностью, а лед — на половину своего объема. Определите объем льда V_l . Определите окончательное изменение уровня глицерина в сосуде ΔH (по отношению к начальному уровню), когда весь лед растает.

Решение

Из условия плавания плотика

$$\rho_p V_p g = \rho_g S \Delta h g$$

определим объем плотика:

$$V_p = \frac{S \Delta h \rho_g}{\rho_p},$$

$$V_p = 900 \text{ см}^3.$$

Из условия плавания плотика и льда

$$\rho_p V_p g + \rho_l V_l g = \rho_g (V_p + 0,5 V_l) g$$

определим объем льда:

$$V_l = \frac{V_p (\rho_g - \rho_p)}{\rho_l - 0,5 \rho_g},$$

$$V_l = 2400 \text{ см}^3.$$

При таянии льда образуется вода плотности $\rho_{\text{в}} = 1,0 \text{ г/см}^3$. Объем образовавшейся воды

$$V_{\text{в}} = \frac{\rho_{\text{л}} V_{\text{л}}}{\rho_{\text{в}}} = 2160 \text{ см}^3.$$

Так как плотность воды меньше плотности глицерина, то вся вода в сосуде расположится выше уровня глицерина. Так как объем воды больше объема плотика, а плотность плотика меньше плотности воды, то плотик будет плавать только в воде.

Таким образом, окончательное изменение уровня глицерина

$$\Delta H = 0 \text{ см.}$$

Ответ: $V_{\text{п}} = 900 \text{ см}^3$, $V_{\text{л}} = 2400 \text{ см}^3$, $\Delta H = 0 \text{ см.}$

Критерии оценивания

| | |
|--|------------------|
| $\rho_{\text{п}} V_{\text{п}} g = \rho_{\text{г}} S \Delta h g$ | 3 балла |
| $V_{\text{п}} = \frac{S \Delta h \rho_{\text{г}}}{\rho_{\text{п}}}$ | 3 балла |
| $V_{\text{п}} = 900 \text{ см}^3$ | 2 балла |
| $\rho_{\text{п}} V_{\text{п}} g + \rho_{\text{л}} V_{\text{л}} g = \rho_{\text{г}} (V_{\text{п}} + 0,5 V_{\text{л}}) g$ | 4 балла |
| $V_{\text{л}} = \frac{V_{\text{п}} (\rho_{\text{г}} - \rho_{\text{п}})}{\rho_{\text{л}} - 0,5 \rho_{\text{г}}}$ | 3 балла |
| $V_{\text{л}} = 2400 \text{ см}^3$ | 2 балла |
| При таянии льда образуется вода плотности $\rho_{\text{в}} = 1,0 \text{ г/см}^3$ | 1 балл |
| $V_{\text{в}} = \frac{\rho_{\text{л}} V_{\text{л}}}{\rho_{\text{в}}} = 2160 \text{ см}^3$ | 2 балла |
| Так как плотность воды меньше плотности глицерина, то вся вода в сосуде расположится выше уровня глицерина; так как объем воды больше объема плотика, а плотность плотика меньше плотности воды, то плотик будет плавать только в воде | 2 балла |
| $\Delta H = 0 \text{ см}$ | 3 балла |
| Всего | 25 баллов |

4.2.3. Физика. 10–11 классы

Задача 4.2.3.1. Мячик (15 баллов)

Тема: кинематика ускоренного движения.

Условие

Мячик бросили с воздушного шара, находящегося в покое на высоте H , со скоростью $v_0 = 20 \text{ м/с}$ под углом $\alpha = 30^\circ$ к горизонту. Через время $t = 2 \text{ с}$ мячик ударился о землю. Определите, с какой скоростью v и под каким углом β к горизонту мячик упал на землю. На какой высоте H находился воздушный шар?

Решение

Предположим, что мячик бросили под углом к горизонту вверх. Уравнения движения мячика по вертикальной оси:

$$y = H + v_0 \sin \alpha t - gt^2/2.$$

В момент падения мячика на землю $y = 0$, таким образом:

$$H = gt^2/2 - v_0 \sin \alpha t = 0,$$

что противоречит условию задачи.

Следовательно, мячик бросили под углом к горизонту вниз. Уравнения движения мячика по вертикальной оси

$$y = H - v_0 \sin \alpha t - gt^2/2.$$

Высота, с которой бросили мячик

$$H = gt^2/2 + v_0 \sin \alpha t = 40 \text{ м.}$$

Проекция скорости мячика при падении на землю на горизонтальную ось

$$v_x = v_0 \cos \alpha \approx 17,3 \text{ м/с.}$$

Проекция скорости мячика при падении на землю на вертикальную ось

$$v_y = -v_0 \sin \alpha - gt = -30 \text{ м/с.}$$

Скорость падения мячика на землю

$$v = (v_x^2 + v_y^2)^{1/2} \approx 34,6 \text{ м/с.}$$

Угол падения мячика на землю

$$\operatorname{tg} \beta = |v_y|/v_x \approx 1,73 \quad (\beta = 60^\circ).$$

Примечание. Скорость падения мячика на землю можно найти из закона сохранения энергии $mv_0^2/2 + mgH = mv^2/2 \rightarrow v = (v_0^2 + 2gH)^{1/2} \approx 34,6 \text{ м/с.}$

Ответ: $H = 40 \text{ м}$, $v \approx 34,6 \text{ м/с}$, $\operatorname{tg} \beta \approx 1,73$ ($\beta = 60^\circ$).

Критерии оценивания

| | |
|---|---------|
| $y = H + v_0 \sin \alpha t - gt^2/2$ | 2 балла |
| $H = gt^2/2 - v_0 \sin \alpha t = 0$, что противоречит условию задачи | 1 балл |
| $y = H - v_0 \sin \alpha t - gt^2/2$ | 2 балла |
| Высота, с которой бросили мячик $H = gt^2/2 + v_0 \sin \alpha t = 40 \text{ м}$ | 2 балла |

| | |
|---|------------------|
| $v_x = v_0 \cos \alpha \approx 17,3 \text{ м/с}$ | 1 балл |
| $v_y = -v_0 \sin \alpha - gt = -30 \text{ м/с}$ | 1 балл |
| $v = (v_x^2 + v_y^2)^{1/2} \approx 34,6 \text{ м/с}$ | 3 балла |
| $\operatorname{tg} \beta = v_y /v_x \approx 1,73 (\beta = 60^\circ)$ | 3 балла |
| Всего | 15 баллов |

Задача 4.2.3.2. Заряды и треугольник (15 баллов)

Темы: электростатика, напряженность электрического поля.

Условие

Точечные заряды $+q_1$ и $-q_2$ поместили в вершины прямоугольного треугольника, лежащих на его гипотенузе. Заряд $-q_2$ находится в вершине треугольника, угол которого равен $\alpha = 30^\circ$. При каком отношении модулей зарядов q_2/q_1 напряженность поля в третьей вершине треугольника будет направлена параллельно гипотенузе?

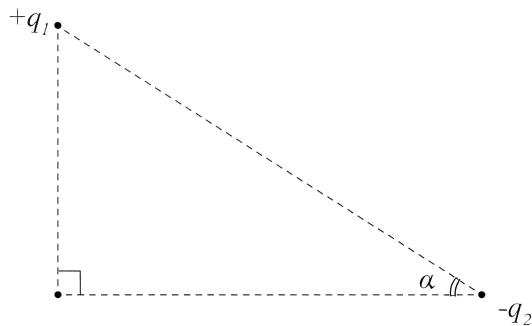


Рис. 4.2.4

Решение

Обозначим: c — гипотенуза прямоугольного треугольника. Поле, создаваемое зарядом $+q_1$, в вершине прямого угла

$$E_1 = kq_1/(c \sin \alpha)^2.$$

Поле, создаваемое зарядом $-q_2$, в вершине прямого угла

$$E_2 = kq_2/(c \cos \alpha)^2.$$

Используя принцип суперпозиции $\vec{E} = \vec{E}_1 + \vec{E}_2$, определим напряженность электрического поля в вершине прямого угла.

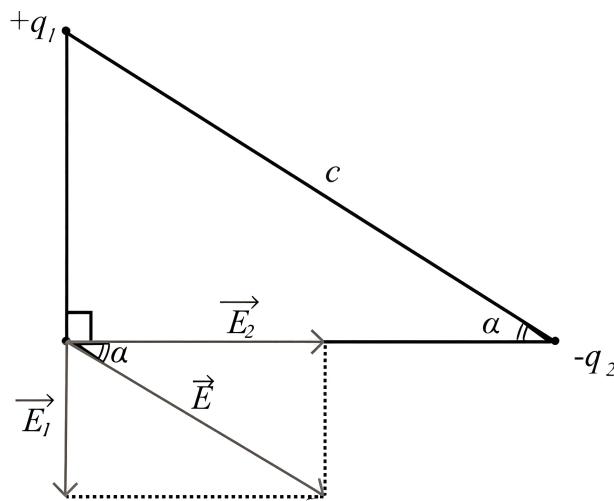


Рис. 4.2.5

Так как треугольник прямоугольный и поле \vec{E} направлено параллельно гипотенузе, то $\tan \alpha = E_1/E_2$.

Из уравнения $\tan \alpha = E_1/E_2 = (q_1 \cos^2 \alpha)/(q_2 \sin^2 \alpha) = (q_1/q_2) \cot^2 \alpha$ получим

$$q_2/q_1 = \cot^3 \alpha = 3\sqrt{3} \approx 5,2.$$

Ответ: $q_2/q_1 = 3\sqrt{3} \approx 5,2$.

Критерии оценивания

| | |
|---|------------------|
| $E_1 = kq_1/(c \sin \alpha)^2$ | 2 балла |
| $E_2 = kq_2/(c \cos \alpha)^2$ | 2 балла |
| $\vec{E} = \vec{E}_1 + \vec{E}_2$ | 3 балла |
| $\tan \alpha = E_1/E_2$ | 2 балла |
| $\tan \alpha = E_1/E_2 = (q_1 \cos^2 \alpha)/(q_2 \sin^2 \alpha) = (q_1/q_2) \cot^2 \alpha$ | 3 балла |
| $q_2/q_1 = \cot^3 \alpha = 3\sqrt{3} \approx 5,2$ | 3 балла |
| Всего | 15 баллов |

Задача 4.2.3.3. Нелинейные элементы (20 баллов)

Темы: закон Ома для участка цепи, мощность.

Условие

Две нелинейных элемента имеют вольтамперные характеристики $U_1 = \alpha_1 I^2$ и $U_2 = \alpha_2/I^2$, где $\alpha_1 = 300 \text{ В/А}^2$, $\alpha_2 = 0,03 \text{ В}\cdot\text{А}^2$. Элементы соединили последовательно и подключили к источнику с напряжением U .

- При каком напряжении U_a источника по цепи идет ток $I_a = 200 \text{ мА}$?

2. При каком минимальном напряжении U_{\min} в цепи может течь ток?
3. Определите мощности, выделяющиеся на элементах при $U = U_{\min}$.

Решение

При последовательном соединении элементов в цепи для общего тока и общего напряжения имеем

$$\begin{aligned} I &= I_1 = I_2; \\ U &= U_1 + U_2; \\ U_a &= \alpha_1 I_a^2 + \alpha_2 / I_a^2; \\ U_a &= 12,75 \text{ В.} \end{aligned}$$

Из уравнения $U = \alpha_1 I^2 + \alpha_2 / I^2$ получим биквадратное уравнение для тока $\alpha_1 I^4 - UI^2 + \alpha_2 = 0$.

Напряжение будет минимальным, если уравнение будет иметь один корень, то есть дискриминант будет равен нулю:

$$\begin{aligned} D &= U^2 - 4\alpha_1\alpha_2 = 0; \\ U_{\min} &= (4\alpha_1\alpha_2)^{1/2} = 6 \text{ В}; \\ I_{\min} &= (U/(2\alpha_1))^{1/2} = 100 \text{ мА.} \end{aligned}$$

Мощность, выделяющаяся на нелинейных элементах:

$$\begin{aligned} P_1 &= I_1 U_1 = \alpha_1 I_{\min}^3 = 300 \text{ мВт}; \\ P_2 &= I_2 U_2 = \alpha_2 / I_{\min} = 300 \text{ мВт.} \end{aligned}$$

Ответ: $U_a = 12,75 \text{ В}$, $U_{\min} = 6 \text{ В}$, $P_1 = 300 \text{ мВт}$, $P_2 = 300 \text{ мВт}$.

Критерии оценивания

| | |
|---|------------------|
| $I = I_1 = I_2$ | 2 балла |
| $U = U_1 + U_2$ | 2 балла |
| $U_a = \alpha_1 I_a^2 + \alpha_2 / I_a^2$ | 2 балла |
| $U_a = 12,75 \text{ В}$ | 2 балла |
| $\alpha_1 I^4 - UI^2 + \alpha_2 = 0$ | 2 балла |
| $D = U^2 - 4\alpha_1\alpha_2 = 0$ | 2 балла |
| $U_{\min} = (4\alpha_1\alpha_2)^{1/2} = 6 \text{ В}$ | 4 балла |
| $I_{\min} = (U/(2\alpha_1))^{1/2} = 100 \text{ мА}$ | 2 балла |
| $P_1 = I_1 U_1 = \alpha_1 I_{\min}^3 = 300 \text{ мВт}$ | 2 балла |
| $P_2 = I_2 U_2 = \alpha_2 / I_{\min} = 300 \text{ мВт}$ | 2 балла |
| Всего | 20 баллов |

Задача 4.2.3.4. Грузики на блоке (25 баллов)

Темы: динамика, второй закон Ньютона.

Условие

Грузики массами $m_1 = m$ и $m_2 = 2m$ прикрепили к концам невесомой нерастяжимой нити, перекинутой через блок массой $M = 3m$. К блоку прикладывают силу $F = 8,5mg$, направленную вертикально вверх. Определите в установившемся режиме движения силу натяжения нити, ускорения грузиков и блока. Ускорение свободного падения g .

Решение

Обозначим: T — сила натяжения нити, a_1 , a_2 и a — ускорения грузов и блока соответственно направленные вертикально вверх.

Запишем уравнения движения блока и грузов:

$$\begin{aligned} Ma &= F - Mg - 2T; \\ m_1a_1 &= T - m_1g; \\ m_2a_2 &= T - m_2g. \end{aligned}$$

Так как блок связан с грузами нерастяжимой нитью, то его ускорение связано с ускорениями грузов:

$$a = (a_1 + a_2)/2.$$

Решая систему записанных уравнений, получим:

$$\begin{aligned} T &= 4/17 \cdot F = 2mg; \\ a &= g/2; \\ a_1 &= g; \\ a_2 &= 0. \end{aligned}$$

Ответ: $T = 2mg$, $a_1 = g$, $a_2 = 0$, $a = g/2$.

Критерии оценивания

| | |
|---------------------|---------|
| $Ma = F - Mg - 2T$ | 3 балла |
| $m_1a_1 = T - m_1g$ | 3 балла |
| $m_2a_2 = T - m_2g$ | 3 балла |
| $a = (a_1 + a_2)/2$ | 4 балла |
| $T = 4/17F = 2mg$ | 3 балла |
| $a = g/2$ | 3 балла |
| $a_1 = g$ | 3 балла |

| | |
|-----------|------------------|
| $a_2 = 0$ | 3 балла |
| Всего | 25 баллов |

Задача 4.2.3.5. Замкнутый процесс (25 баллов)

Тема: молекулярная физика.

Условие

Идеальный газ участвует в замкнутом процессе $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 1$. На участках $1 \rightarrow 2$ и $3 \rightarrow 4$ температура газа пропорциональна квадрату объема, а процессы $2 \rightarrow 3$ и $4 \rightarrow 1$ изотермические. Найдите максимальное давление p_{\max} газа в цикле, если известно, что минимальное давление $p_{\min} = 1$ атм, объемы в состояниях 2 и 4 равны $V_2 = 60$ дм³ и $V_4 = 30$ дм³, а максимальный объем $V_{\max} = 90$ дм³.

Решение

Уравнение состояния идеального газа

$$pV = \nu RT.$$

Для процессов, в которых температура газа пропорциональна квадрату объема ($T = \alpha V^2$) получим

$$pV = \nu R \alpha V^2 \rightarrow p = \alpha \nu R V = \beta V$$

(давление пропорционально объему).

Рассмотрим процесс в осях pV .

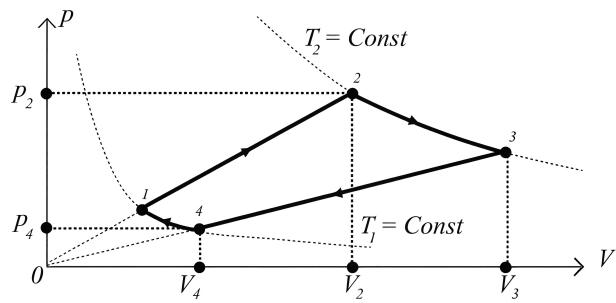


Рис. 4.2.6

Исходя из данных задачи, имеем

$$p_4 = p_{\min},$$

$$p_2 = p_{\max},$$

$$V_3 = V_{\max}.$$

Так как на участке $3 \rightarrow 4$ давление пропорционально объему, то

$$p_3/p_4 = V_3/V_4 \rightarrow p_3 = p_4 V_3/V_4 = p_{\min} V_{\max}/V_4.$$

Так как участок $2 \rightarrow 3$ — изотермический, то

$$p_2 V_2 = p_3 V_3 \quad (\text{или } p_{\max} V_2 = p_3 V_{\max})$$

Окончательно получим

$$p_{\max} = p_3 V_{\max} / V_2 = p_{\min} (V_{\max})^2 / (V_2 V_4),$$

$$p_{\max} = 4,5 \text{ атм.}$$

Ответ: $p_{\max} = 4,5$ атм.

Критерии оценивания

| | |
|---|------------------|
| $p_4 = p_{\min}$ | 2 балла |
| $p_2 = p_{\max}$ | 2 балла |
| $V_3 = V_{\max}$ | 2 балла |
| $p_3/p_4 = V_3/V_4 \rightarrow p_3 = p_4 V_3/V_4 = p_{\min} V_{\max}/V_4$ | 4 балла |
| $p_2 V_2 = p_3 V_3 \quad (\text{или } p_{\max} V_2 = p_3 V_{\max})$ | 4 балла |
| $p_{\max} = p_3 V_{\max}/V_2 = p_{\min} (V_{\max})^2 / (V_2 V_4)$ | 3 балла |
| $p_{\max} = 4,5 \text{ атм}$ | 2 балла |
| Всего | 25 баллов |

4.3. Инженерный тур

4.3.1. Общая информация

Участникам необходимо разработать микро-ТНПА (управляемый подводный аппарат с одним движителем и цифровой камерой), предназначенный для выполнения осмотровых работ при диагностике состояния морского судна, который может перемещаться в ограниченных пространствах и выполнять видеосъемку. Кроме того, нужно создать алгоритм классификации обрастаний на дне судна и обнаружения повреждений на основе полученного видеопотока.

Какую реальную проблему решают участники

Участники решают проблему автоматизации процесса проведения диагностических работ в реальных условиях. Они разрабатывают инструмент для проведения диагностических работ под водой и выполняют обработку результатов с применением методов компьютерного зрения.

4.3.2. Легенда задачи



Рис. 4.3.1

Компании N необходимо устройство для экспресс-диагностики состояния судов, так как не всегда есть возможность привлечения водолазов для этих работ. Поэтому решено внедрить подводных роботов в качестве альтернативного способа диагностики.

В настоящий момент основными проблемами компании являются:

- анализ состояния дна морского судна;
- количество обрастаний на днище;

- исследования ограниченных областей в донной части, куда сложно зайти в полевых условиях.

Задача

Разработать микро-ТНПА, который будет иметь в своей конструкции цифровую камеру и независимый пульт управления. С помощью аппарата-носителя MiddleROV необходимо будет доставить микро-ТНПА до места исследования, определить тип обрастваний на корпусе морского судна. Затем с помощью микро-ТНПА нужно будет:

1. провести исследование подводного шлюза;
2. выявить повреждения;
3. подсчитать количество повреждений (отверстий) в нем.

Порядок выполнения задания:

1. Разработать **микро-ТНПА**.
2. Разработать **конструкцию** микро-ТНПА (а именно, способ фиксации камеры в колбе, способ крепления движителя к колбе микро-ТНПА, способ транспортировки).
3. Разработать **программу управления** микро-ТНПА с помощью пульта.
4. Разработать **программу для обработки видеопотока** с микро-ТНПА: программу для классификации обрастваний на судне и определения повреждений в трубе.
5. **Выполнить миссию.** В рамках миссии необходимо исследовать корпус судна, определив тип обраствания на нем, зайти в подводный шлюз и определить повреждения.

4.3.3. Требования к команде и компетенциям участников

Рекомендуемый состав команды — 4 участника:

1. **конструктор**: изготавливает корпус микро-ТНПА, крепление движителя, способ транспортировки микро-ТНПА с помощью MiddleROV;
2. **электронщик**: собирает электрическую схему, припаивает компоненты на плату, разъем;
3. **программист микроконтроллеров**: настраивает передачу данных между пультом управления и микро-ТНПА, настраивает управление мотором с помощью пульта;
4. **программист компьютерного зрения**: пишет ПО для распознавания и классификации повреждений, определения отверстий в трубе.

4.3.4. Оборудование и программное обеспечение

Все электромонтажные и конструкторские работы выполняются в специальной рабочей зоне.

В рабочей зоне расположены необходимые инструменты и оборудования для осуществления электромонтажа (паяльные станции, расходные материалы, источники питания и т. д.) и для выполнения работ по конструированию (режущие инструменты, шлифовальные, измерительные и т. д.). В рабочей зоне размещается стол с дополнительными комплектующими и крепежом, которые могут быть использованы для решения задачи на усмотрение участников.

Таблица 4.3.1

| Наименование | Описание |
|---|--|
| Arduino IDE https://www.arduino.cc/en/software | Для программирования глубиномера |
| САПР (например, SolidWorks и др.) | Для создания чертежа общего вида/принципиальной схемы |
| MUR IDE https://murproject.com/#muride (рекомендуется) | Для отработки задания по компьютерному зрению |
| Паяльные станции (6 шт.) | Для пайки электронной части микро-ТНПА |
| Набор ручного инструмента (4 набора) | Для выполнения работ по разработке микро-ТНПА |
| ТНПА MiddleROV | Аппарат-носитель для транспортировки микро-ТНПА и выполнения миссии. |

- Командам необходимо привезти с собой свои ноутбуки для работы (не менее двух).
- Команда может привезти с собой собственный набор ручных инструментов, который сочтет достаточным для выполнения задания.
- Командам запрещено использовать собственные комплектующие для выполнения задания.

Для выполнения задания заключительного этапа каждая команда получает набор комплектующих от разработчиков профиля. Не рекомендуется приступать к использованию компонентов до объявления задания.

Команда полностью несет ответственность за сохранность своего набора. Все комплектующие прошли проверку и работоспособны. В случае, если команда повредила камеру, плату, драйверы, движитель или плату с микроконтроллером замена не предоставляется.

Таблица 4.3.2. Состав набора

| № | Наименование | Кол-во |
|------------------------|--|--------|
| Гермоконтейнер | | |
| 1 | Купол (55 мм) | 1 |
| 2 | Крышка фланцевая передняя акрил | 1 |
| 3 | Крышка фланцевая задняя ПНД | 1 |
| 4 | Гильза | 1 |
| 5 | Кольцо прижимное алюминиевое | 2 |
| 6 | Шпильки резьбовые | 2 |
| 7 | Гайка М3 | 4 |
| 8 | Уплотнительное кольцо $45 \times 2,5$ мм | 1 |
| 9 | Кольцо уплотнительное силиконовое красное $OD = 60$, $cs = 3,5$ | 2 |
| 10 | Герметичный кабельный ввод на 4 контакта M12-H4-02-SS | 1 |
| 11 | Кабельный ввод M6 + гайка | 1 |
| 12 | Разъем двигателя Chasing Dory + стопорное кольцо | 1 |
| 13 | Комплект крепежа | Общий |
| Электроника | | |
| 1 | Драйвер E-MAX Bullet 12A | 1 |
| 2 | Микросхема MAX3232 | 1 |
| 3 | Модуль камеры CCDCAM IM50A01 IMX335 Gk7205 1,7 мм | 1 |
| 4 | Комплект крепежа камеры (винт M2×10 + гайка M2 нейлон) × 4 | 1 |
| 5 | Шлейф обжатый для подключения к камере Ethernet 10 Pin | 1 |
| 6 | Arduino Nano (ATmega328P) | 1 |
| 7 | Плата макетная | 1 |
| 8 | Конденсаторы керамические | Общий |
| 9 | Разъем/гребенка | Общий |
| 10 | Набор резисторов | Общий |
| 11 | Стабилизатор DC-DC Boost до 12В | 1 |
| Комплектующие | | |
| 1 | Двигатель Chasing Dory правый (5) или левый (1) | 1 |
| 2 | Комплект крепежа для двигателя: (винт M2×12 + гайка M2) × 3 | 1 |
| 3 | Кабель плоский шлейф 4 жилы | 10 м |
| 4 | Кабель Ethernet CAT 6A 32AWG (2,8 мм), 20м | 10 м |
| 5 | Разъем ответный для пульта управления | 1 |
| 6 | Изолента | 1 |
| Надводная часть | | |
| 1 | Джойстик (комплект с панелями) | 1 |

| № | Наименование | Кол-во |
|----------------------------|---|------------|
| 2 | Комплект крепежа (стойки $M2 \times 8$ н-в, $M2 \times 8$ в-в, 2 винта $M2 \times 6$) $\times 6$ | 1 |
| 3 | Литиевый аккумулятор 18650 | 2 |
| 4 | Переходник Ethernet-USB | По запросу |
| 5 | Зарядное устройство для аккумуляторов | Общее |
| Расходные материалы | | |
| | Провода | Общее |
| | Фомакс | Общее |
| | АКП (алюкобонд) | Общее |
| | Свинец для балластировки | Общее |
| | Плавучесть | Общее |
| | Стержни для клеевого пистолета | Общее |

4.3.5. Описание задачи

Задание на разработку микро-ТНПА

Микро-ТНПА — герметичное устройство с одним двигателем, способное перемещаться в толще воды и выполнять видеосъемку под водой, транслируя изображение с камеры на поверхность. Управление микро-ТНПА осуществляется с помощью пульта, связанного с подводной частью кабелем.

1. Наименование выполняемых работ

Разработка микро-ТНПА для выполнения диагностики состояния морского судна.

2. Требования к устройству:

2.1. Требования к микро-ТНПА:

- Поверхностная часть состоит из пульта управления с аккумуляторами и кабеля.
- Подводная часть устройства состоит из колбы с платой и камерой, двигателя.
- Корпус устройства должен позволять ему свободно перемещаться в ограниченных пространствах (трубах) диаметром от 110 мм.
- Длина корпуса (без учета кабеля) не должна быть больше 20 см.
- Аппарат должен иметь возможность движения вперед и назад.
- Питание устройства осуществляется с джойстика (пульт управления микро-ТНПА) с помощью двух аккумуляторов 18650.
- Камера должна быть жестко зафиксирована внутри корпуса.
- Для двигателя должно быть разработано его крепление к колбе.
- Запись и обработка видеопотока осуществляются с цифровой камеры микро-ТНПА. Вывод изображения должен осуществляться на ноутбук.

бук напрямую по кабелю Ethernet.

- Необходимо отбалластировать микро-ТНПА. Аппарат должен иметь плавучесть, близкую к нулевой.
- Управление микро-ТНПА должно осуществляться с пульта управления. Распиновка пульта представлена в [Приложении 1](#).
- Связь между пультом и микро-ТНПА осуществляется с помощью интерфейса RS232.
- Светодиоды должны работать согласно регламенту: при нажатии на кнопку **Вперед** — светодиод горит зеленым, **Назад** — красным. Команда может задействовать либо 1, либо 2 светодиода на пульте управления для реализации этой функции.

2.2. Крепление мотора и камеры:

- Мотор должен быть прикреплен к задней крышке аппарата и/или к стягивающим шпилькам. Мотор присоединен к микро-ТНПА устойчиво.
- Вывод проводов осуществляется через заднюю крышку аппарата через герметичные разъемы.
- Камера жестко закреплена внутри корпуса. Отсутствует смещение камеры при движении аппарата.
- Все кабели выведены через герметичные разъемы.
- Для крепления движителя к корпусу запрещено использовать термо-клей и изоленту в качестве единственного способа крепления. Эти материалы допускается применять в качестве вспомогательных.
- Со списком компонентов можно ознакомиться в разделе «Материалы для изготовления».

2.3. Способ транспортировки микро-ТНПА:

- Доставку микро-ТНПА к месту проведения работ осуществляет подводный аппарат MiddleROV.
- Микро-ТНПА можно переносить в манипуляторе аппарата.
- Команда получает дополнительные баллы, если разработает способ транспортировки микро-ТНПА не в манипуляторе MiddleROV.
- В качестве точки крепления на раме MiddleROV есть крепежный уголок, который может быть использован командами в качестве точки крепления микро-ТНПА для транспортировки.

3. Требования к ПО:

3.1. ПО для управления микро-ТНПА:

- Состоит из двух частей: прошивка для пульта и прошивка для платы микро-ТНПА.
- Связь между пультом и подводной частью осуществляется с помощью интерфейса RS232.
- При нажатии и удерживании кнопки **Вверх** — аппарат должен двигаться вперед, при нажатии и удерживании кнопки **Вниз** — назад.
- Распиновка пульта управления представлена в [Приложении 1](#).

3.2. ПО для обработки видеопотока с камеры микро-ТНПА

ПО для обработки видеопотока должно решать следующие задачи:

Задача 1. Классификация обрастаний на стенке судна.

Задача 2. Определение и подсчет повреждений в трубе.

Инструкция по работе с цифровой камерой представлена в справочном документе: <https://disk.yandex.ru/i/eEi8mTgn5Wp9Qg>.

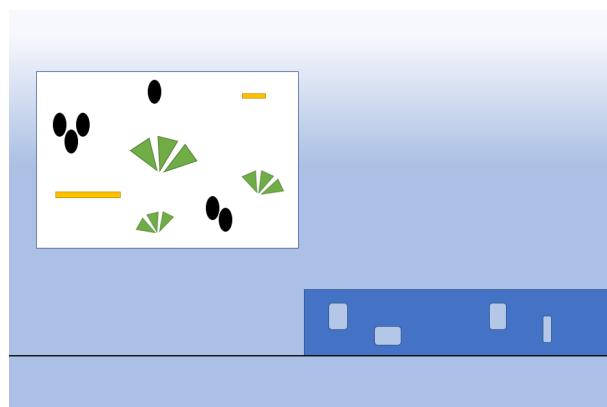


Рис. 4.3.2. Схема миссии

Задача 1. Основными типами обрастаний, которые могут быть зафиксированы на корпусе судна, являются:

- ◊ **Двустворчатые моллюски.** Представляют собой группы черных овалов. Количество овалов может быть от одного до трех. Размеры моллюсков неизменны, может отличаться только их количество в группе. Команде нужно выделить группу моллюсков (но не каждый моллюск по отдельности!). В примере ниже приведены три группы моллюсков.
- ◊ **Водоросли.** Представляют собой группу из трех зеленых треугольников. Количество и положение треугольников в группе неизменно. Отличаться может только их размер (см. рис. ниже).
- ◊ **Области коррозии.** Прямоугольники оранжевого цвета. Ширина прямоугольников неизменна, может меняться только их длина.

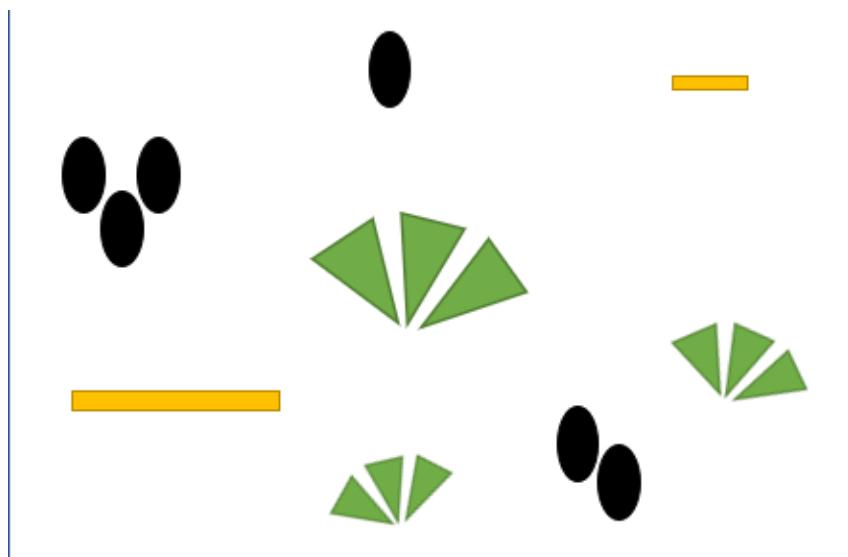


Рис. 4.3.3. Типы обрастаний на корпусе судна

Классификация обрастаний:

- ◊ При погружении в воду программист компьютерного зрения запускает программу на своем ноутбуке.
- ◊ На мониторе должно быть два окна: видеопоток, получаемый с аппарата, и обработанное видео (на обработанном видео происходит определение типа обрастаний и их подсчет).
- ◊ При движении в толще воды аппарат должен совершить остановку напротив таблички, обозначающей стенку морского судна. При этом на мониторе программиста компьютерного зрения должно быть видно изображение всей таблички. Необходимо выделить все найденные обрастания и подписать их тип.
- ◊ В левом верхнем углу необходимо указать, сколько обрастаний и какого типа было обнаружено.
- ◊ В тот момент, когда программист компьютерного зрения готов сдать задачу, он говорит: «Готов!» и нажимает на клавишу 1.
- ◊ После нажатия на клавишу 1 на мониторе создается новое окно с изображением, на котором видна табличка с выделенными объектами и с их определенным количеством. Табличка с наростами должна находиться полностью в кадре. В верхнем левом углу кадра должна быть таблица с указанием типов наростов и их количеством.
- ◊ Необходимо предусмотреть сохранение этого изображения.
- ◊ В рамках попытки команда может выполнить фиксацию результата классификации только один раз.

Пример

- ◊ Shellfish: 3.
- ◊ Algae: 3.
- ◊ Corrosion: 2.

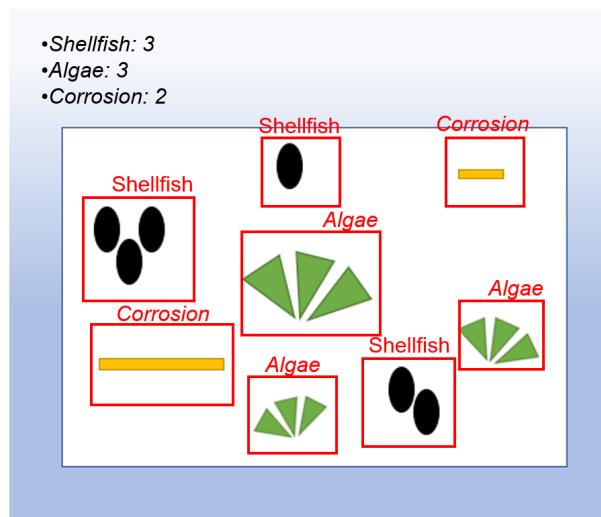


Рис. 4.3.4

Задача 2. Определение и подсчет повреждений на трубе:

- ◊ Необходимо заснять прохождения микро-ТНПА по трубе, моделирующей подводный шлюз и выполнить обработку видео в ре-

альном времени.

- ◊ Необходимо в процессе движения микро-ТНПА по трубе, выделять контуры всех обнаруженных отверстий (повреждений) и вести их подсчет, отражая количество обнаруженных повреждений на видео.
- ◊ Примечание: отверстие в конце трубы тоже считается за повреждение.
- ◊ Порядок выполнения задания:
 - * Пилот ТНПА доставляет микро-ТНПА до трубы и оповещает судью о готовности приступить к работе.
 - * После этого на мониторе должно быть два окна: видеопоток с аппарата (микро-ТНПА) и обработанное видео (на обработанном видео происходит выделение повреждений, а в верхнем углу должно выводиться количество этих повреждений). Подробнее в разделе [Порядок проведения за-плыва](#).

4. Состав работ

Таким образом, командой должны быть выполнены следующие работы:

- 4.1. Разработка электроники микро-ТНПА, удовлетворяющего требованиям.
- 4.2. Разработка конструкции микро-ТНПА, удовлетворяющей требованиям.
- 4.3. Создание комплекта конструкторской документации:
 - Набор конструкторской документации (чертеж общего вида, чертежи трех конструкций: крепление камеры, крепление двигателя, крепление микро-ТНПА к ТНПА MiddleROV).
 - Принципиальная электрическая схема устройства.
- 4.4. Разработка ПО для управления микро-ТНПА.
- 4.5. Разработка ПО для проведения работ по классификации объектов.

5. Результат работы:

5.1. Собранный микро-ТНПА

Собранный микро-ТНПА, который функционирует согласно заявленным требованиям. ПО для обработки видеопотока с микро-ТНПА, которое позволяет производить диагностику состояния морского судна.

5.2. Принципиальная электрическая схема

Принципиальная электрическая схема микро-ТНПА (подводная часть). Принципиальная схема оформляется согласно [критериям оценки](#). Название файла должно быть в формате [микроТНПА_название_команды.pdf](#) (пример, [микроТНПА_MUR.pdf](#)).

5.3. Комплект конструкторской документации

Должен состоять из чертежа общего вида микро-ТНПА, спецификации и чертежей разработанных конструкций: крепления камеры, крепления двигателя, способа транспортировки ТНПА. Название файла должно быть в формате [корпус_название_команды.pdf](#) (пример, [корпус_MUR.pdf](#)).

Команда должна объединить чертежи и спецификацию в один файл

и прислать только его. Если от команды будут получены два файла, то рассматриваться будет последний присланный.

5.4. Программа для управления микро-ТНПА

Всего должно быть загружено два файла: код для пульта (надводной части) и код для ТНПА (подводной части). Код для подводной и надводной части должны быть сложены в папки с названием `команда_bottom` и `команда_surface` и объединены в архив с названием `команда_microrov.zip` (архивы других форматов или просто загруженный код рассматриваться не будут и команда получает 0 баллов и не допускается до заплыков).

Например, для команды MUR решение будет выглядеть так: архив `MUR_microrov.zip`, а в нем две папки: `mur_bottom` и `mur_surface`.

5.5. Программа для обработки изображения с камеры

Состоит из двух частей: задание 1 (определение обрастаний) и задание 2 (определение повреждений в трубе).

Коды для подводной и надводной части должны быть сложены в папки с названием `команда_task1` и `команда_task2` и объединены в архив с названием `команда_opencv.zip` (архивы других форматов или просто загруженный код рассматриваться не будут и команда получает 0 баллов и не допускается до заплыков).

Например, для команды MUR решение будет выглядеть так: архив `MUR_opencv.zip`, а в нем две папки: `mur_task1` и `mur_task2`.

6. Сдача работы

6.1. Результаты работы проходят оценку в три этапа:

- Оценка документации (+ проверка на плагиат).
- Демонстрация работы на воздухе.
- Демонстрация работы в воде.

6.2. Вся документация (чертежи, электрическая принципиальная схема) должна быть загружена в форму по ссылке до 20 февраля 2025 18:30.

6.3. Программа для управления микро-ТНПА должна быть загружена в гугл-форму по ссылке. Дедлайн: 21 февраля 2025 10:00 (по местному времени).

6.4. Программа для обработки изображения с камеры должна быть загружена в форму по ссылке. Дедлайн: 21 февраля 2025 10:00.

6.5. *Если документацию команда присыпает после дедлайна, то такие работы не рассматриваются и оцениваются в 0 баллов.*

6.6. Итоговая демонстрация работы на воздухе проходит 20 февраля, согласно установленному расписанию. Команда может представить конструкцию для проверки на воздухе **один раз** досрочно до 19 февраля 18:30. Если команду не устроил результат, участники могут забрать конструкцию на доработку и повторно сдать ее в день итоговой демонстрации на воздухе.

6.7. Оценка соответствия конструкции предоставленным чертежам проходит 21 февраля перед началом заплыков, после сдачи конструкций в каран-

тин.

- 6.8. Итоговая демонстрация работы микро-ТНПА в воде проводится 21 февраля.
- 6.9. Подробнее о процессе сдачи работы в разделе [Порядок проведения соревнований и оценивания](#).

Приложение 1. Пульт управления

Ниже представлена распиновка пульта управления, который идет в комплекте с набором для конструирования микро-ТНПА. Питание устройства осуществляется с помощью двух аккумуляторов 18650.

Таблица 4.3.3

| Arduino Pin Name | Buoy Pin Function | | MiddleBuoy Gamepad |
|------------------|-------------------|---|----------------------|
| D7 | LED Strip PWM | + | |
| +5V | Power | | |
| RD- | USB | + | USB-Serial to PC |
| RD+ | USB | + | |
| UGND | Ground | | |
| UCAP | USB Cap | | |
| VBus | USB VBUS | | |
| RXLED | RX LED | + | Serial Status LED |
| SCK | ICSP | + | Debug Port |
| MOSI | ICSP | + | |
| MISO | ICSP | + | |
| D11 (PWM) | Input + PullUp | + | Button Down |
| RESET | ICSP | + | Debug Port |
| +5V | Power | | |
| GND | Ground | | |
| XTAL2 | OSC Output | | |
| XTAL1 | OSC Input | | |
| D3 (SCL)(PWM) | I2C SCL | + | OLED Display SSD1306 |
| D2 (SDA) | I2C SDA | + | |
| D0 (RX) | UART RX | + | |
| D1 (TX) | UART TX | + | |
| TXLED | TX LED | + | |
| GND | Ground | | |
| AVCC | Analog Power | | |

| | | | |
|-----------|----------------|---|-------------------------------|
| D4 | Input + PullUp | + | Button Up |
| D12 | Input + PullUp | + | |
| D6 (PWM) | LED Strip PWM | + | WS2812B |
| D8 | IO | | |
| D9 (PWM) | Servo PWM | + | |
| D10 (PWM) | Servo PWM / IO | | |
| D5 | Input + PullUp | + | |
| D13 (PWM) | PWM / Output | + | Status LED (Boot) |
| HWB | Boot Mode | | См. схематик Arduino Leonardo |
| +5V | Power | | |
| GND | Ground | | |
| A0 | Analog Input | + | |
| A1 | Analog Input | + | Voltmeter |
| A2 | Analog / IO | | |
| A3 | Analog / IO | | |
| A4 | Analog / IO | | |
| A5 | Analog / IO | | |
| AEF | Analog Ref | | |
| GND | Ground | | |
| AVCC | Analog Power | | |

4.3.6. Система оценивания

1. Критерии оценки приведены в разделе [Критерии оценивания](#).
2. Документация оценивается двумя судьями — в зачет идет среднее значение оценок судей.
3. Демонстрация работы на воздухе проходит в конце тренировочного дня согласно расписанию.
4. Команда имеет право пройти проверку на воздухе досрочно. В случае, если команда не согласна с результатами проверки, она может попытаться устраниТЬ недочеты в своей конструкции и пройти проверку заново согласно общему расписанию.
5. Проверка соответствия изделия предоставленным чертежам проводится перед заплывами 21 февраля.
6. Демонстрация работы в воде проходит в испытательном бассейне. У команды две попытки на демонстрацию работы. В зачет идет лучшая попытка.

Порядок проведения испытаний и оценивания результатов работы

Общая информация

1. Все устройства и разработанный программный код проходят проверку на воздухе и проверку в воде.
2. Команда может пройти проверку на воздухе раньше установленного срока согласно расписанию.
3. В таком случае судья проводит проверку результата и оглашает итоговый балл за проверку. Если команду не устраивает оценка, то она может забрать свое устройство на доработку и пройти проверку согласно расписанию. В зачетную таблицу идут результаты **последней** проверки.
4. Разработчики профиля подготавливают расписание тренировок. Порядок тренировок команд определяется жеребьевкой 19 февраля в конце дня. Тренировочный день — 20 февраля.
5. Финальные заплывы проводятся 21 февраля.
6. Перед началом каждой попытки устанавливается табличка с обрастваниями и труба, положение которых не меняется в течение всего заплыва для всех команд.
7. Перед началом попыток **все** разработанные микро-ТНПА сдаются в карантин. Ноутбуки остаются на столах и работать за ними в период проведения заплывов **запрещено**.
8. Изображения на таблице и количество повреждений в трубе на финальных заплывах отличаются от предоставленных макетов для тренировок в пределах, указанных в регламенте.
9. Каждой команде предоставляется две попытки по 15 мин. В рамках одной попытки команда имеет право выполнить заплыв один раз.
10. В рамках попытки предусмотрено три этапа: установка микро-ТНПА, демонстрация в воде (заплыв), извлечение микро-ТНПА из воды.

Порядок проведения заплыва

- Команде дается 15 мин на выполнение задания.
- Команда подходит на рабочую станцию со своим ноутбуком и микро-ТНПА согласно расписанию заплывов и извещает судью о готовности начать заплыв.
- Судья запускает таймер 15 мин.
- За это время команда должна:
 - ◊ Установить микро-ТНПА в захват подводного аппарата MiddleROV.
 - ◊ Выполнить демонстрацию работы в бассейне.
 - ◊ Извлечь микро-ТНПА из воды.
- При завершении попытки капитан команды говорит: «Стоп!», и судья останавливает таймер, фиксирует время и набранные баллы.
- Команда может завершить попытку и сказать «стоп» только тогда, когда микро-

ТНПА будет извлечен из воды и доставлен на бортик бассейна.

- Извлечь микро-ТНПА команды могут вручную.
- **Алгоритм проверки определения типа обрастаний:**

- ◊ При погружении в воду программист компьютерного зрения запускает программу на своем ноутбуке.
- ◊ На мониторе должно быть два окна: видеопоток, получаемый с аппарата и обработанное видео (на обработанном видео происходит определение типа обрастаний и их подсчет).
- ◊ В тот момент, когда программист компьютерного зрения готов сдать задачу, он говорит: «Готов!» и нажимает на клавишу 1.
- ◊ После нажатия на клавишу 1 на мониторе создается новое окно с изображением, на котором видна табличка с выделенными объектами и с их определенным количеством. Табличка с наростами должна находиться полностью в кадре. В верхнем левом углу кадра должна быть таблица с указанием типов наростов и их количеством (см. п. 3.2).
- ◊ Необходимо предусмотреть сохранение этого изображения.
- ◊ Судья оценивает результат работы по изображению в третьем окне, после чего аппарат переходит к выполнению следующей задачи.

- **Алгоритм проверки прохождения трубы:**

- ◊ Пилот ТНПА доставляет микро-ТНПА до трубы и оповещает судью о готовности приступить к работе.
- ◊ После этого на мониторе должно быть два окна: видеопоток с аппарата (Микро-ТНПА) и обработанное видео (на обработанном видео происходит выделение повреждений, а в верхнем углу должно выводиться количество этих повреждений).
- ◊ Микро-ТНПА должен пройти через всю трубу, т. е. достичь второго конца трубы с заглушкой: судья видит на мониторе, что изображение на камере не меняется, и расстояние до отверстия в крышке остается неизменным, а при нажатии кнопки **Вперед** робот не смещается (движение отсутствует).
- ◊ После фиксации судьей прохождения трубы микро-ТНПА должен выплыть из нее своим ходом (**запрещено тянуть** кабель микро-ТНПА, с целью его вытащить).
- ◊ Судья фиксирует завершения обследования в момент выхода микро-ТНПА из трубы, и сразу после этого команда может извлечь устройство за кабель.

- **Завершение попытки**

- ◊ Команда может завершить попытку после громкой команды «стоп» только при выполнении следующих условий:
 - * ТНПА MiddleROV возвращен к точке старта и извлечен из воды на бортик бассейна.
 - * Микро-ТНПА извлечен из воды.
- ◊ Если команда выполнила условия завершения заплыва, а затем была дана команда «стоп», то судья останавливает таймер.

- ◇ После остановки таймера производится проверка конструкции на сохранение герметичности и целостности после заплыва.

Штрафные баллы

- Если команда не уложилась в отведенное время выполнения задания, то фиксируются баллы, набранные на момент окончания времени миссии, также начисляется 5 штрафных баллов.
- Нарушение техники безопасности: первое нарушение — предупреждение, второе и последующие: -5 баллов.
- Обfuscация (запутывание кода)/плагиат — баллы за задачу обнуляются у всех команд, у которых он был обнаружен.
- Если члены команды тянут ТНПА (любой и за любой из трех кабелей), то за первое нарушение — предупреждение, за второе второе и последующие — вычитается 5 баллов.
- Если команда тянет за кабели микро-ТНПА с целью спозиционировать его или вытащить из трубы, то баллы за критерий следования и возврата микро-ТНПА по трубе команда получает 0 баллов и предупреждение. При повторном нарушении: -5 баллов. Разрешено тянуть за кабель микро-ТНПА только с целью его извлечения из воды, когда он полностью вышел из трубы.

Критерии оценивания

| № | Критерий | Балл | Пояснение |
|---|--|------|---------------------------------|
| Документация | | | |
| Принципиальная схема. Данная схема разрабатывается в соответствии с ГОСТ 2.702-2011. | | | |
| 1 | Наличие заполненной рамки для заглавного листа в соответствии с ГОСТ 2.104-2006 | 1 | |
| 2 | Линии связи состоят из горизонтальных и вертикальных отрезков и не имеют изломов и взаимных пересечений | 1 | |
| 3 | Все надписи выполнены шрифтом ГОСТ тип Б | 1 | |
| 4 | Схема снабжена перечнем элементов с указанием их номиналов | 2 | |
| 5 | Обозначение компонентов соответствует УГО | 2 | |
| 6 | Схема соответствует условию задачи: 0 — не соответствует, 1 — частично соответствует, 2 — полностью соответствует | 2 | |
| Чертеж общего вида (разрабатывается в соответствии с ГОСТ 2.120-73) | | | |
| 1 | Рамка и основная надпись на заглавной и последующих страницах соответствует ГОСТ | 1 | Количество листов не ограничено |

| № | Критерий | Балл | Пояснение |
|---|---|------|--|
| 2 | Представлены изометрия и минимум 2 вида (разрез/вырыв считается за вид, а местный вид или вид с разрывом — нет) | 2 | Речь про чертеж всей конструкции микро-ТНПА. |
| 3 | Указаны габаритные размеры всей сборки и каждой разрабатываемой конструкции отдельно | 2 | Разрабатываемые конструкции: крепление камеры, крепление движителя, крепление микро-ТНПА к ТНПА |
| 4 | Чертеж дополнен спецификацией с рамкой по ГОСТ, разрабатываемые детали и готовые явно отделены, указано корректное количество и номера соответствуют указателям на чертеже | 2 | Детали из индивидуального набора считать прочими изделиями; для указания всех деталей рекомендуется применить взрыв-схему или использовать достаточное для указания всех деталей конструкции количество видов и различные их типы. |
| 5 | Качество исполнения чертежа: 0 — нечитаемо, не соответствует требованиям, 1 — есть вопросы по оформлению, частично соответствует, 2 — 1–2 незначительные ошибки, соответствует требованиям | 2 | |

Код

| | | | |
|---|--|---|--|
| 1 | Программа для микро-ТНПА загружена вовремя | 1 | |
| 2 | Программа для микро-ТНПА загружена вовремя | 1 | |

Проверка на воздухе**Конструкция**

| | | | |
|---|---|---|---|
| 1 | Конструкция соответствует предоставленным чертежам | 1 | |
| 2 | В конструкции нет потенциальной опасности (острых выступов и деталей) | 2 | |
| 3 | Крепление движителя реализовано и соответствует требованиям | 2 | Мотор должен быть прикреплен к задней крышке аппарата; в креплении используется не только термоклей; изолента, применяется по необходимости |
| 4 | Камера надежно зафиксирована внутри корпуса | 2 | |
| 5 | Аппарат имеет плавучесть близкую к нулевой | 2 | |
| 6 | Конструкция аппарата позволяет переносить его не в манипуляторе ТНПА | 4 | |

Электроника + МК

| | | | |
|---|---|---|--|
| 1 | Электроника соответствует предоставленной схеме | 2 | |
| 2 | В исполнении схемы нет потенциальной опасности | 3 | |

| № | Критерий | Балл | Пояснение |
|------------------------|---|------|---|
| 3 | Включение устройства осуществляется с помощью пульта управления | 2 | |
| 4 | Светодиоды на пульте работают согласно регламенту | 2 | |
| 5 | Мотор работает (при нажатии кнопки Вверх — движение вперед, при нажатии кнопки Вниз — движение назад) | 3 | |
| 6 | Видеопоток с камеры выводится на монитор ноутбука | 2 | |
| Машинное зрение | | | |
| 1 | Контуры объектов определяются | 1 | |
| 2 | Реализована функция подписи типа объектов | 1 | |
| 3 | Отверстия в трубе распознаются на воздухе | 1 | |
| Проверка в воде | | | |
| 1 | Микро-ТНПА остался герметичен после проверки в воде (визуальное наличие воды в колбе) | 3 | Шаг считается успешно выполненным, если после завершения миссии судья фиксирует отсутствие влаги внутри корпуса микро-ТНПА |
| 2 | Конструкция микро-ТНПА надежна (после выполнения миссии все части микро-ТНПА остались на месте) | 3 | Шаг считается успешно выполненным, если после завершения миссии судья фиксирует отсутствие деталей от микро-ТНПА и креплений в бассейне и целостность конструкции |
| 3 | Микро-ТНПА выполнил исследование всей трубы: 10 баллов — выполнил полностью, 5 баллов — частично выполнено, 0 баллов — не выполнено | 10 | Шаг считается успешно выполненным если микро-ТНПА доплыл до конца трубы и на ноутбуке видно изображение бассейна через отверстие в конце трубы |
| 4 | Возврат микро-ТНПА | 5 | Шаг считается успешно выполненным, если микро-ТНПА достиг противоположной стороны трубы, а затем вернулся в стартовую точку; микро-ТНПА полностью вышел из трубы. |
| 5 | Получено изображение корпуса судна и трубы | 5 | Шаг считается выполненным, если видеопоток с аппарата транслируется на мониторе и в кадр попало изображение судна и трубы |
| 6 | Правильно выделены все типы обрастаний: 10 баллов — 8 наростов выделены верно, 8 баллов — от 5 до 7 наростов выделены верно, 4 балла — от 2 до 4 наростов выделены верно, 0 баллов — от 0 до 1 нароста выделены верно | 10 | Шаг считается выполненным, если каждый нарост выделен в отдельную рамку и рамка подписана соответствующим названием |

| № | Критерий | Балл | Пояснение |
|--------------|---|------------|---|
| 7 | Посчитано количество обрастваний разного типа на корпусе судна | 4 | Шаг считается выполненным, если на изображении с выделенными наростами сверху есть таблица с указанием количества наростов разного типа и их количество указано корректно |
| 8 | Определены контуры и подсчитано количество повреждений: 2 балла — определено 1 повреждение, 4 балла — выделено 2–3 повреждения, 6 баллов — выделено 4 повреждения, 8 баллов — выделено 5 и более повреждений, 10 баллов — выделены все повреждения и их количество | 10 | Шаг считается выполненным, если при движении микро-ТНПА по трубе на видеопотоке выделяются отверстия и ведется их подсчет |
| ИТОГО | | 100 | |

Порядок начисления баллов и определения победителя

- В итоговый зачет идут баллы, набранные командой за лучшую попытку в воде из двух.
- Результатом оценки документации является среднее значение оценок судей.
- Результатом оценки на воздухе является балл набранный при последней проверке.
- После завершения оценок формируется единый рейтинг. Команда-победитель определяется по сумме баллов за все испытания. Если две команды набрали одинаковое количество баллов, то в расчет берется результат худшей попытки.
- Индивидуальный результат определяется по формуле: Итоговый балл = 20% физика + 20% информатика + 60% инженерный тур.

4.3.7. Решение задачи

1. Разработка микро-ТНПА

Разработка электроники микро-ТНПА. Для выполнения задачи участникам необходимо собрать схему подводной части аппарата на основе Arduino Nano, а также наладить связь между пультом и микро-ТНПА с использованием интерфейса RS232. На плате пульта микросхема для реализации связи уже установлена, поэтому участникам нужно было использовать микросхему MAX3232.

Примеры реализации микро-ТНПА см. рис. 4.3.5–4.3.7.



Рис. 4.3.5. Команда «Керосин»



Рис. 4.3.6. Команда ROZEN



Рис. 4.3.7. Команда «Кокосы»

Для разработанного микро-ТНПА необходимо было составить принципиальную электрическую схему: рис. 4.3.8.

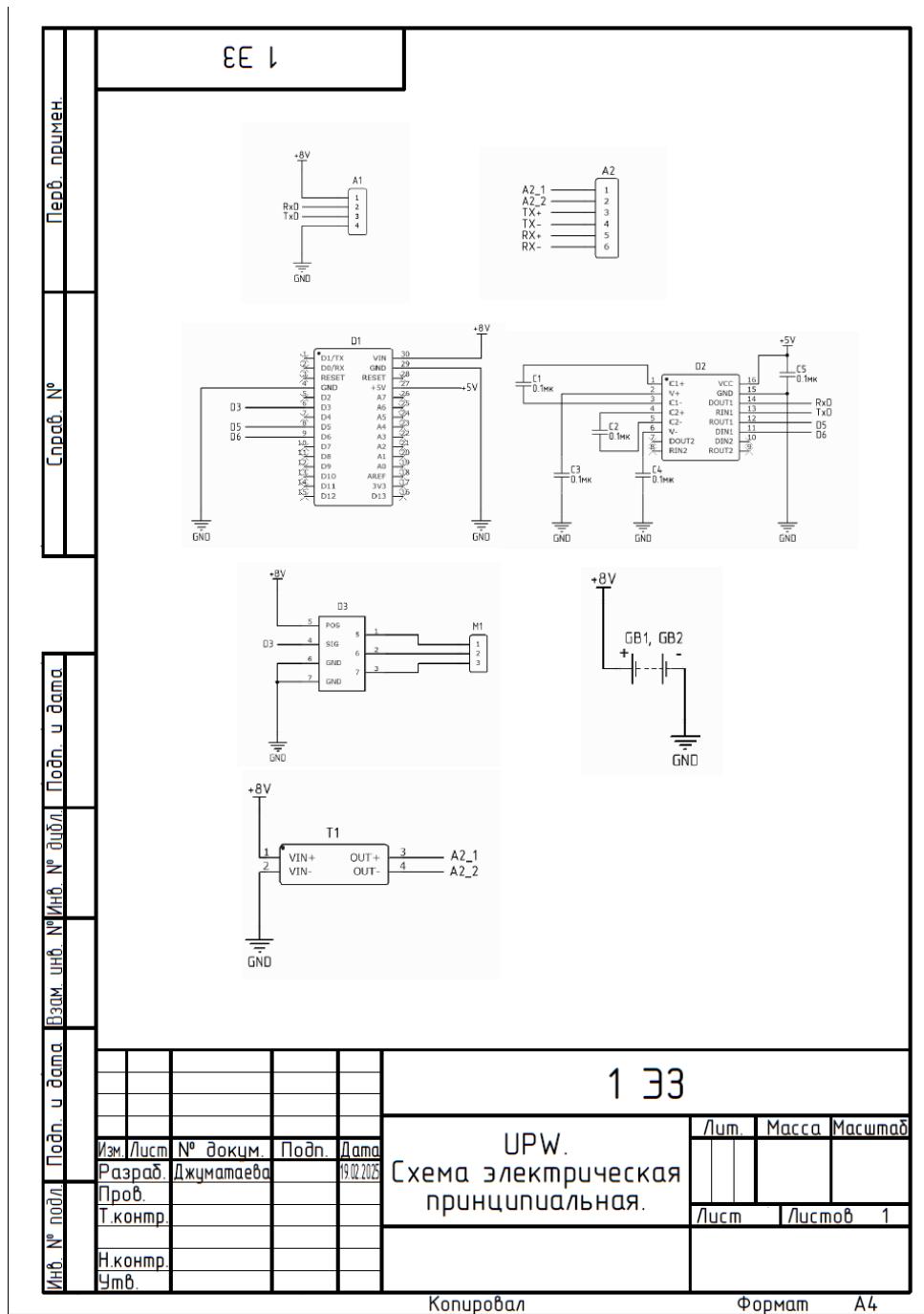


Рис. 4.3.8. Пример принципиальной схемы команды UPW (8 из 8 баллов)

Примеры принципиальных схем: <https://disk.yandex.ru/d/0WoMzE3yBu8SKA>.

2. Разработка корпуса и способа транспортировки

Результатом работы является набор конструкторской документации (чертеж общего вида, чертежи трех конструкций: крепление камеры, крепление движителя, крепление микро-ТНПА к ТНПА MiddleROV). Микро-ТНПА и способ транспор-

тировки проходят тестирование на воздухе и в воде. В качестве точки крепления микро-ТНПА к ТНПА мог быть использован уголок, установленный в нижней части аппарата или отверстия M5 в нижней пластине рамы ТНПА.

Примеры выполнения конструкторской документации: рис. 4.3.9.

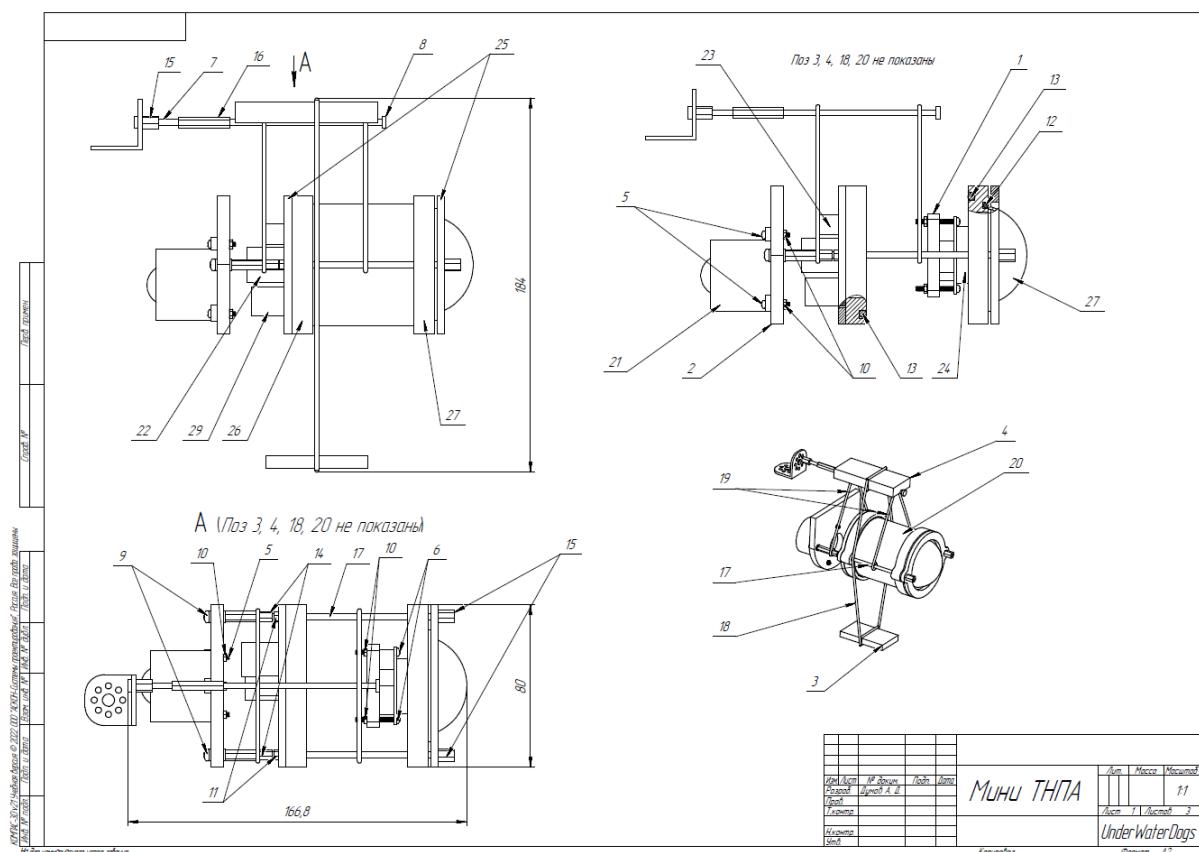


Рис. 4.3.9. Микро-ТНПА команды Underwater Dogs (9 из 9 баллов)

Пример комплекта конструкторской документации: <https://disk.yandex.ru/d/Wc1zhQ1GMPSajA>.

3. ПО управления микро-ТНПА

Для успешного решения поставленной задачи программисту микроконтроллеров необходимо написать код для надводной (пульт) и подводной (микро-ТНПА) частей аппарата. При работе микро-ТНПА должна выполняться следующая последовательность действий:

- При нажатии на кнопку **Вперед** светодиод должен светиться зеленым цветом, при нажатии на кнопку **Назад** — красным.
- Для управления движителем, необходимо подать на управляющий контакт ШИМ (PWM) сигнал со следующими характеристиками:
 - Частота 50 Гц.
 - Ширина импульса от 1 000 до 2 000 мкс:
 - * Сигналу 1 000 мкс соответствует полный назад (тяга -100%).
 - * Сигналу 1 500 мкс соответствует стоп (тяга 0%).
 - * Сигналу 2 000 мкс соответствует полный вперед (тяга +100%).

Указанные диапазоны могут варьироваться для каждого мотора (к примеру, нейтральному положению может соответствовать сигнал 1 485 мкс, а не 1 500 мкс). Участникам может понадобиться самостоятельно подобрать значение, соответствующее нулевой тяге. Для успешной инициализации при запуске на мотор следует подавать сигнал с нулевой тягой в течение нескольких секунд.

Ссылка на архив с кодом с комментариями от разработчиков: <https://disk.yandex.ru/d/Vs3sGsaS7rFfbQ/nto-camera.zip>.

4. ПО для проведения работ по классификации объектов

Для успешного решения поставленной задачи программисту компьютерного зрения необходимо обработать видео с цифровой камеры следующим образом:

1. Распознать значение глубины на глубиномере и вывести его.
2. Распознать фигуры на табличке и выделить их.
3. Определить класс фигур на табличке и подписать каждую.
4. Посчитать количество объектов каждого класса и вывести результат на экран.
5. По нажатию на кнопку 1 вывести статичную картинку с результатом.

Для подзадачи 2 необходимо:

1. Распознать круглые отверстия в трубе.
2. Исключить возможность распознавания отражений в трубе в качестве отверстия
3. Настроить вывод количества обнаруженных отверстий в обработанном видеопотоке.

Примеры реализации кода от участников финала («Кокосы» — 50 за миссию, ROZEN — 50 за миссию): <https://disk.yandex.ru/d/WS6DpLc2bgrJew>, <https://disk.yandex.ru/d/GtUw104hhU13hg>.

4.3.8. Материалы для подготовки

- MUR IDE: <https://murproject.com/#muride>.
- Arduino IDE: <https://www.arduino.cc/en/software>.
- Практикум профиля Водные робототехнические системы: <https://disk.yandex.ru/i/NAhopKofwZNhwQ>.
- Видео по подводной робототехнике: <https://vk.com/murproject>.

5. Критерии определения победителей и призеров

Первый отборочный этап

В первом отборочном этапе участники решали задачи предметного тура по двум предметам: информатике и физике 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 туре не менее 100,0 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 222,0 баллов, независимо от уровня.

Заключительный этап

Индивидуальный предметный тур

- информатика — максимально возможный балл за все задачи — 100 баллов;
- физика — 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 (информатика) с весом $K_1 = 0,2$, по предмету

2 (физика) с весом $K_2 = 0,2$, баллы за командное решение задач инженерного тура с весом $K_3 = 0,6$.

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 информатике (предметный тур) ($S_{1 \text{ макс}} = 100$);

S_2 — балл первой части заключительного этапа по физике (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 3 победителя и 6 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 | 68,50 и выше |
| Призеры | От 60,30 до 66,50 |

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 (Р2Р). 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».