

Департамент образования и науки Курганской области
Государственное автономное нетиповое образовательное учреждение
Курганской области
«Центр развития современных компетенций»

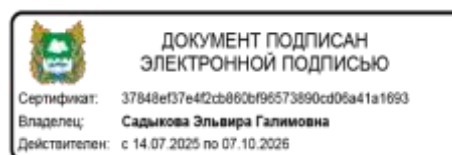
ПРИНЯТА

на заседании педагогического
совета
от «29» августа 2025 г.

Протокол № 1

УТВЕРЖДАЮ

И.о директора ГАНБОУ КО «ЦРСК»
_____ Садыкова Э.Г.
Приказ от «29» августа 2025 г. №441



ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ ПРОГРАММА
технической направленности
«Информационные технологии» нулевая линия

Ознакомительный уровень
Возраст учащихся: 12-18 лет
Срок реализации: 18 недель

Автор-составитель:

Бобовский Георгий Евгеньевич,
педагог дополнительного образования

Курган, 2025

ПАСПОРТ ПРОГРАММЫ

Ф.И.О. автора/авторов	Бобовский Георгий Евгеньевич
Учреждение	ГАНОУ КО ЦРСК ДТ «Кванториум»
Наименование программы	«Информационные технологии» нулевая линия
Квантум	IT-квантум
Тип программы	Дополнительная общеобразовательная программа
Направленность программы	Техническая
Образовательная область	Информационные технологии
Вид программы	Модифицированная
Продолжительность реализации программы	18 недель
Возраст учащихся	12-18 лет
Объем часов по годам обучения	72 часа
Уровень программы	ознакомительный уровень
Цель программы	знакомство с теоретическими и практическими основами знаний в области программирования через решение кейсов.
С какого года реализуется программа	2023 г.

СОДЕРЖАНИЕ

РАЗДЕЛ 1. КОМПЛЕКС ОСНОВНЫХ ХАРАКТЕРИСТИК ПРОГРАММЫ	3
1.1. Пояснительная записка.....	3
1.2. Цель и задачи программы. Планируемые результаты.....	5
1.4. Рабочая программа.....	7
РАЗДЕЛ 2. КОМПЛЕКС ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИХ УСЛОВИЙ	11
2.1. Календарный учебный график.....	11
2.2. Формы контроля.....	12
2.3. Материально – техническая база.....	12
2.4. Методические материалы.....	13
2.5. Список литературы и интернет – источников.....	15
Приложение 1.....	16

РАЗДЕЛ 1. КОМПЛЕКС ОСНОВНЫХ ХАРАКТЕРИСТИК ПРОГРАММЫ

1.1. Пояснительная записка

Дополнительная общеобразовательная программа «Информационные технологии» нулевая линия технической направленности составлена с учетом следующих документов:

- Федеральный закон «Об образовании в Российской Федерации» № 273-ФЗ от 29.12.2012 года (с изменениями, внесенными Федеральным законом от 31.07.2020 № 304-ФЗ «О внесении изменений в Федеральный закон «Об образовании в РФ по вопросам воспитания обучающихся» ст. 2 п. 9; с изменениями, вступившими в силу 25.07.2022;

- Распоряжение Правительства Российской Федерации от 31.03.2022 г. № 678-р «Об утверждении Концепции развития дополнительного образования детей до 2030 г. и плана мероприятий по ее реализации»;

- Приказ Министерства просвещения Российской Федерации от 27 июля 2022 г. № 629 «Об утверждении порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам»;

- Постановление Главного санитарного врача РФ от 28.09.2020 года № 28 «Об утверждении санитарных правил СП 2.4.3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи»;

- Письмо Минобрнауки России от 18.11.2015 г. № 09-3242 «О направлении информации» (вместе с «Методическими рекомендациями по проектированию дополнительных общеразвивающих программ (включая разноуровневые программы)»);

- Приказ Министерства труда и социальной защиты Российской Федерации от 22 сентября 2021 года № 652н «Об утверждении профессионального стандарта «Педагог дополнительного образования детей и взрослых» (Зарегистрировано в Минюсте России 17.12.2021 N 66403);

- Государственная Программа Курганской области «Развитие образования и реализация государственной молодежной политики» (с изменениями на 31 января 2019 года);

- Письмо Департамента образования и науки Курганской области от 26.10.2021 г. № 08-05794/21 “ О структурной модели дополнительной общеобразовательной программы”;

- Устав Государственного автономного нетипового образовательного учреждения Курганской области «Центр развития современных компетенции», Лицензией учреждения, Правилами внутреннего трудового распорядка учреждения, Программой развития, Локальными актами и иными нормативно-правовыми документами учреждения;

- Положение о дополнительных общеобразовательных программах Государственного автономного нетипового образовательного учреждения Курганской области “Центр развития современных компетенции” (приказ №571 от 16.09.2024).

Актуальность программы

Программа «Информационные технологии» нацелена на погружение в современное программирование с нуля. Программирование – главное в развитии компьютерных технологий. Понимание процесса программирования позволяет взглянуть на мир информационных технологий «изнутри», приводит к осознанию устройства и архитектуры современных компьютерных программ и приложений.

Изучая программирование, обучающиеся учатся анализировать и решать задачи, писать чистый код, понимать алгоритмы и структуры данных, создавать программы и тестировать написанный код. Помимо этого, дети развивают системное и творческое мышление, а также навыки решения текущих задач и проблемных ситуаций.

Первый язык программирования часто определяет скорость и эффективность освоения новой профессии. Python – высокоуровневый язык программирования общего назначения с простым синтаксисом, который обеспечит быстрое обучение и поможет развить алгоритмическое мышление. Его проще учить в сравнении с другими языками, поэтому он отлично подходит для начинающих.

Python гарантирует нам легкий старт, позволяет «погрузиться» в мир информационных технологий. Но для того, чтобы продолжить наш путь в IT-индустрии, нам необходимо определиться с направлением. Существует множество различных направлений в сфере IT, каждое из которых имеет свою специфику, но все они объединены общей целью – использованием технологий для решения проблем и улучшения жизни людей. Вот некоторые из ключевых областей в IT: веб-разработка, разработка программного обеспечения, мобильная разработка, безопасность информационных технологий, аналитика данных, облачные технологии, IT-консалтинг и многое другое.

Наиболее актуальным и востребованным направлением в IT-индустрии всегда была и остается веб-разработка. Веб-разработка включает в себя создание веб-сайтов или веб-приложений. Это направление подразделяется на фронтенд (клиентская сторона) и бэкенд (серверная сторона) разработку.

Мы же остановимся на фронтенд-разработке. Фронтенд-разработка фокусируется на создании интерфейса, который пользователи видят при посещении веб-сайта. Фронтенд-разработчики используют технологии, такие как HTML, CSS и JavaScript, чтобы создать визуальное представление веб-сайта.

HTML – это язык разметки, который мы используем для визуального и смыслового структурирования нашего web-контента, например, определяем параграфы, заголовки, таблицы данных, или вставляем изображения и видео на страницу.

CSS – это язык стилей с помощью которого мы придаём стиль отображения нашего HTML контента, например придаём цвет фону (background) и шрифту, придаём контенту многоколоночный вид.

JavaScript – это язык программирования, который позволяет вам создать динамически обновляемый контент, управляет мультимедиа, анимирует изображения, впрочем, делает всё, что угодно.

Отличительные особенности программы

Программа разбита на отдельные кейсы, каждый кейс сопровождается минимальным теоретическим введением и несколькими десятками задач. Подбор задач выполняется исходя из умений, навыков и теоретической подготовки обучающихся к программированию. Задачи каждому выдаются индивидуально, задачи посильные, т.е. заведомо ориентированы на успех.

Адресат программы

Программа ориентирована на обучающихся среднего и старшего школьного возраста (12-18 лет) проявляющих интерес к информационным технологиям, имеющих высокую мотивацию к обучению и хорошие знания по математике, информатике и английскому языку.

Срок реализации (освоения) программы: 18 недель

Объем программы: 72 часа

Формы обучения, особенности организации образовательного процесса – групповая, с элементами индивидуальной работы. Занятия проводятся очно, допускается временное дистанционное обучение по причине отмены занятий по погодным условиям и с введением карантинных мер, обусловленных высокой заболеваемостью среди детей и педагогов.

Образовательный процесс (занятия) осуществляется в группах обучающихся разного возраста. Состав групп постоянный; количество обучающихся регламентируется требованием СанПиН и локальных актов образовательной организации.

Комплектование учебных групп осуществляется по личному заявлению родителей (законных представителей). Доукомплектование осуществляется в течение всего учебного года при наличии вакантных мест в ГАНОУ КО «Центр развития современных компетенций», структурное подразделение «Детский технопарк «Кванториум» по направлению IT-квантум.

Режим занятий: 2 раза по 2 часа в неделю.

Продолжительность занятия: 2 академических часа.

Структура двухчасового занятия:

- 45 минут (рабочая часть)
- 10 минут (перерыв)
- 40 минут (рабочая часть)
- 5 минут (рефлексия)

Возможность реализации индивидуального образовательного маршрута (ИОМ) предусматривается и проектируется из индивидуальных потребностей обучающихся.

Наличие детей инвалидов и детей с ограниченными возможностями здоровья (ОВЗ) предусматривается и проектируется из индивидуальных потребностей обучающихся.

Наличие талантливых детей в объединении. Для данной категории обучающихся предусматривается и проектируется ИОМ по сопровождению проектов повышенной сложности по темам программы.

Уровни сложности содержания программы: программа направлена на получение знаний умений и навыков стартового (ознакомительного) уровня в области программирования.

1.2. Цель и задачи программы. Планируемые результаты.

Цель: знакомство с теоретическими и практическими основами знаний в области программирования через решение кейсов.

Задачи программы

Обучающие:

- познакомить алгоритмизацией и логикой;
- познакомить с базой данных и базовыми принципами работы сети интернет;
- познакомить с возможностями языков программирования Python, JavaScript;

- познакомить с основами языков гипертекстовой разметки HTML и таблиц стилей CSS.

Воспитывающие:

- воспитывать самоорганизацию и ответственность;
- формировать бережное отношение к технике;
- формировать чувство коллективизма и взаимопомощи;
- формировать мотивацию к профессиональному самоопределению обучающихся, приобщению к социально-значимой деятельности для осмысленного выбора профессии.

Развивающие:

- формировать чувство ответственности, внутренней инициативы, самостоятельности, тяги к самосовершенствованию и мотивацию к творческому поиску информации;
- развивать алгоритмическое мышление у обучающихся;
- формировать интерес к техническим знаниям;
- формировать у обучающихся умение работать в команде.

1.3. Планируемые результаты

При реализации программы «Информационные технологии. Нулевая линия» обучающиеся достигнут следующих результатов:

Личностные (Soft Skills):

- умение генерировать идеи указанными методами;
- умение слушать и слышать собеседника;
- умение аргументировать свою точку зрения;
- умение искать информацию и структурировать ее;
- умение работать в команде;
- самостоятельно выбрать цель собственного развития, пути достижения цели, постановка новых задач в познании;
- соотнесение собственных возможностей и поставленных задач;
- критическое мышление и умение объективно оценивать результаты своей работы;
- навыки ораторского искусства.

Метапредметные:

- умение самостоятельно находить и отбирать информацию для решения учебных задач;
- владение основами самоконтроля, взаимоконтроля и коррекции результатов.

Предметные (Hard Skills):

- знание основ языка программирования Python;
- знание основ алгоритмизации и логики;
- знание основ работы с базами данных;
- знание встроенных и сторонних библиотек Python;
- знание базовых принципов работы сети интернет;
- знание основ языка гипертекстовой разметки HTML;
- знание основ языка таблиц стилей CSS;

- знание основ языка программирования JavaScript;
- знание основ современных принципов объектно-ориентированного программирования и работы с функциями;
- знание основ работы со страницей.

1.4. Рабочая программа. Учебный план. Содержание программы. Тематическое планирование.

№	Название раздела/кейса программы	Количество часов			Формы промежуточной аттестации
		Всего	Теория	Практика	
1	Первые шаги	2	1	1	Наблюдение, опросно-ответная форма
2	Основы	8	2	6	Решение задач кейса
3	Операторы и выражения	6	1	5	Решение задач кейса
4	Поток команд	10	3	7	Решение задач кейса
5	Функции	12	2	10	Решение задач кейса
6	Модули	4	1	3	Решение задач кейса
7	Структуры данных	12	2	10	Решение задач кейса
8	Ввод-вывод	10	2	8	Решение задач кейса
9	Исключения	4	1	3	Решение задач кейса
14	Аттестационные мероприятия	4	0	4	Тестирование. Решение алгоритмических задач. Разработка индивидуального/группового проекта. Презентация проектов.
		72	15	57	

Содержание программы

Раздел 1. Первые шаги. (Количество часов: теория - 1 час, практика - 1 час).

Теория: 1. Техника безопасности. Знакомство с программой обучения. Введение в язык программирования. Знакомство с редактором кода.

Практика: работа с редактором кода.

Раздел 2. Основы. (Количество часов: теория - 2 час, практика - 6 час).

Кейс 1 «Основы» - комментарии. Литеральные константы. Числа. Строки. Логические и физические строки. Отступы. Переменные. Имена идентификаторов. Типы данных. Объекты.

Раздел 3. Операторы и выражения. (Количество часов: теория - 1 час, практика - 5 час).

Кейс 2 «Операторы и выражения» - операторы. Порядок вычисления. Изменение порядка вычисления. Ассоциативность. Выражения.

Раздел 4. Поток команд. (Количество часов: теория - 3 час, практика - 7 час).

Кейс 3. «Оператор if» - оператор while. Цикл for. Оператор break. Оператор continue.

Раздел 5. Функции. (Количество часов: теория - 2 час, практика - 10 час).

Кейс 4. «Функции» - параметры функций. Значения аргументов по умолчанию. Ключевые аргументы. Переменное число параметров. Только ключевые параметры. Оператор «return». Строки документации. Локальные переменные. Зарезервированное слово «global». Зарезервированное слово «nonlocal». Lambda-формы. Работа с lambda-функциями.

Раздел 6. Модули. (Количество часов: теория - 1 час, практика - 3 час).

Кейс 5. «Модули» - файлы байткода .рус. Оператор from ... import ... Имя модуля - `_name_`. Создание собственных модулей. Функция `dir`. Пакеты.

Раздел 7. Структуры данных. (Количество часов: теория - 2 часа, практика - 10 часа).

Кейс 6. «Структуры данных» - список. Краткое введение в объекты и классы. Кортеж. Словарь. Последовательности. Множество. Ссылки. Ещё о строках.

Раздел 8. Ввод-вывод. (Количество часов: теория - 2 часа, практика - 8 часа).

Кейс 7. «Ввод-вывод» - ввод от пользователя. Файлы. Pickle. Оператор `with`.

Раздел 9. Исключения. (Количество часов: теория - 1 час, практика - 3 час).

Кейс 8. «Ошибки» - исключения. Обработка исключений. Try ... Finally.

Раздел 10. Аттестационные мероприятия. (Количество часов: теория - 0 часов, практика - 4 часа).

Выбор формы аттестации определяется из предложенных: тестирование/ решение алгоритмических задач/ разработка индивидуального или группового проекта и его презентация.

Тематическое планирование.

№ п/п	Название раздела программы	Дата проведения занятия	Кол-во часов	Тема занятия	Форма занятия	Форма текущего контроля/ промежуточной аттестации
1	Первые шаги	01.09.25 02.09.25	2	Техника безопасности Знакомство с программой обучения Введение в язык программирования	Лекция, беседа	Наблюдение, опросно-ответная форма

				Знакомство с редактором кода		
2	Основы	03.09.25 04.09.25	2	Кейс 1. «Основы»	Лекция, беседа, практика	Решение задач
		08.09.25 09.09.25	2		Лекция, беседа, практика	Решение задач
		10.09.25 11.09.25	2		Лекция, беседа, практика	Решение задач
		15.09.25 16.09.25	2		Лекция, беседа, практика	Решение задач
3	Операторы и выражения	17.09.25 18.09.25	2	Кейс 2. «Операторы и выражения»	Лекция, беседа, практика	Решение задач
		22.09.25 23.09.25	2		Лекция, беседа, практика	Решение задач
		24.09.25 25.09.25	2		Лекция, беседа, практика	Решение задач
4	Поток команд	29.09.25 30.09.25	2	Кейс 3. «Поток команд»	Лекция, беседа, практика	Решение задач
		01.10.25 02.10.25	2		Лекция, беседа, практика	Решение задач
		06.10.25 07.10.25	2		Лекция, беседа, практика	Решение задач
		08.10.25 09.10.25	2		Лекция, беседа, практика	Решение задач
		13.10.25 14.10.25	2		Лекция, беседа, практика	Решение задач

5	Функции	15.10.25 16.10.25	2	Кейс 4. «Функции»	Лекция, беседа, практика	Решение задач
		20.10.25 21.10.25	2		Лекция, беседа, практика	Решение задач
		22.10.25 23.10.25	2		Лекция, беседа, практика	Решение задач
		27.10.25 28.10.25	2		Лекция, беседа, практика	Решение задач
		29.10.25 30.10.25	2		Лекция, беседа, практика	Решение задач
		03.11.25 04.11.25	2		Лекция, беседа, практика	Решение задач
6	Модули	05.11.25 06.11.25	2	Кейс 5. «Модули»	Лекция, беседа, практика	Решение задач
		10.11.25 11.11.25	2		Лекция, беседа, практика	Решение задач
7	Структуры данных	12.11.25 13.11.25	2	Кейс 6. «Структуры данных»	Лекция, беседа, практика	Решение задач
		17.11.25 18.11.25	2		Лекция, беседа, практика	Решение задач
		19.11.25 20.11.25	2		Лекция, беседа, практика	Решение задач
		24.11.25 25.11.25	2		Лекция, беседа, практика	Решение задач
		26.11.25 27.11.25	2		Лекция, беседа, практика	Решение задач

		01.12.25 02.12.25	2		Лекция, беседа, практика	Решение задач
8	Ввод-вывод	03.12.25 04.12.25	2	Кейс 7. «Ввод-вывод»	Лекция, беседа, практика	Решение задач
		08.12.25 09.12.25	2		Лекция, беседа, практика	Решение задач
		10.12.25 11.12.25	2		Лекция, беседа, практика	Решение задач
		15.12.25 16.12.25	2		Лекция, беседа, практика	Решение задач
		17.12.25 18.12.25	2		Лекция, беседа, практика	Решение задач
9	Исключения	22.12.25 23.12.25	2	Кейс 8. «Исключения»	Лекция, беседа, практика	Решение задач
		24.12.25 25.12.25	2		Лекция, беседа, практика	Решение задач
10	Аттестационные мероприятия	29.12.25 30.12.25	2	Тестирование Решение алгоритмических задач	Практика	Тестирование, решение задач
		13.01.26	2	Разработка индивидуального/группового проекта Презентация проектов	Практика	Индивидуальная/групповая проектная деятельность

РАЗДЕЛ 2. КОМПЛЕКС ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИХ УСЛОВИЙ

2.1 Календарный учебный график

Количество учебных недель	18 недель
---------------------------	-----------

Первое полугодие	с 01.09.2025 г. по 15.01.2026 г.
Промежуточная аттестация	с 23.12.2025 г. по 30.12.2025 г.
Каникулы	с 31.12.2025 г. по 11.01.2026 г.

2.2. Формы контроля

С целью определения уровня усвоения программы, а также для повышения эффективности и улучшения качества учебно-воспитательного процесса проводится аттестация обучающихся в течение всего периода обучения. Форма и время проведения аттестации регламентируется педагогом.

Промежуточный контроль проводится для обучающихся в конце прохождения каждого раздела, кейса на последнем занятии с целью проверки усвоения полученной информации в форме тестирования, решения задач и разработки индивидуального/группового проекта (приложение 1).

Итоговый контроль проводится в рамках промежуточной аттестации для обучающихся. С целью проверки усвоения информации, полученной за полный курс программы. Промежуточная аттестация проводится в форме тестирования, решения задач и разработки индивидуального/группового проекта.

Для перехода на следующую линию обучения, обучающимся необходимо набрать достаточное количество баллов за тестирование и за творческое задание или набрать достаточное количество баллов за участие в конкурсах.

2.3. Материально – техническая обеспечение

Требования к помещению и инфраструктуре:

- подключение к интернету;
- рабочие места.

Оборудование и материалы:

- персональный компьютер/ноутбук — 1 шт. на одного обучающегося;
- проектор с экраном/ТВ с возможностью подключения к ноутбуку – 1 шт.;
- пакет программ для программирования на языке Python;
- пакет программ для работы с базой данных SQLite;
- пакет программ для программирования на языке JavaScript;
- пакет программ для языков HTML, CSS;

Кадровое обеспечение:

Преподавание по программе осуществляет педагог дополнительного образования Бобовский Георгий Евгеньевич, ДТ «Кванториум», соответствующий требованиям профессионального стандарта «Педагог дополнительного образования детей и взрослых» (Приказ Министерства труда и социальной защиты Российской Федерации от 22 сентября 2021 года № 652н «Об утверждении профессионального стандарта «Педагог дополнительного образования детей и взрослых»).

Информационное обеспечение

- https://developer.mozilla.org/ru/docs/Learn/Common_questions/Web_mechanics/How_does_the_Internet_work
- <https://www.youtube.com/watch?v=vAjHdh92YU>
- <https://www.youtube.com/watch?v=bOZqmy9PAOk>
- <https://www.youtube.com/watch?v=0P1gtxRWh-s>
- <https://pythonru.com/osnovy/sqlite-v-python>
- <https://habr.com/ru/articles/442800/>
- <https://github.com/Emberium/SimpleQIWI>
- <https://python-scripts.com/tkinter>
- <https://webref.ru/course>
- <https://learn.javascript.ru/>

2.4. Методические материалы

Методы обучения – при реализации программы используются как традиционные методы: словесный, наглядный, объяснительно-иллюстративный, практический, так и нетрадиционные: частично-поисковый, проблемный, игровой, проектный, кейсовый.

Формы организации образовательной деятельности – занятия организуются с учетом разного уровня подготовки обучающихся, возрастных и гендерных особенностей контингента объединения; предусматривают коллективную, групповую и индивидуальную формы работы.

Формы организации учебного занятия – выбор формы организации учебного занятия зависит от содержания учебного материала, подготовки обучающихся и результата, который должен быть получен по итогам изучения того или иного материала. Диапазон форм, которые могут быть использованы для организации учебного занятия в дополнительном образовании, широк. Остановимся на нескольких, которые представляются нам наиболее целесообразными и эффективными для реализации программы:

- учебное занятие – основная традиционная форма образовательной деятельности, используется педагогом при изучении нового учебного материала, закреплении знаний и способов деятельности, а также при проверке, оценке, коррекции знаний и способов деятельности (если нецелесообразно использовать нетрадиционные формы);

- работа в мини-группах – это методика объединения обучающихся в небольшие группы для совместного выполнения заданий кейсов. Используется для того, чтобы обучающиеся овладели коммуникативными умениями и навыками. Совместная работа развивает умение общаться, слушать, коллективно решать проблемы, достигать взаимопонимания;

- презентация проекта – представление обучающимися результатов своей творческой деятельности в создании игр;

- кейс-метод - одна из основных новых форм обучения в данной программе является проблемно-ситуативное обучение с использованием кейсов.

Кейс-метод – это интерактивная технология для краткосрочного обучения на основе реальных невымышленных ситуаций, направленных не только на усвоение знаний, но и на формирование у обучающихся новых качеств и умений.

Кейс (от англ. case – «случай») — это специально подготовленный учебный материал, который ставит перед обучающимися конкретную проблемную ситуацию, требующую решений, а также ряд источников информации, изучив которые обучающиеся учатся сравнивать, анализировать, применять в действии, создавать продукты (артефакты) и делать выводы.

Для каждого кейса/проекта определен следующий сценарий работы: проблема → цель → задачи → поиск оптимального решения → решение → анализ и оформление продукта (артефакта) → защита проекта.

Результаты выполненных проектов в кейсах должны быть «осязаемыми», то есть, это конкретные продукты (артефакты), готовые к использованию в реальной жизни.

Педагогические технологии:

- технология разноуровневого обучения используется в данной программе для обеспечения усвоения учебного материала на разных уровнях сложности и адаптируется относительно возможностей и темпа развития каждого обучающегося;

- информационно-коммуникационные технологии, в основе которых разнообразные программно-технические средства, используются педагогом для решения определенных образовательных задач, имеющие предметное содержание и ориентированные на взаимодействие с обучающимся;

- технология сотрудничества (обучение во взаимодействии) основана на использовании различных методических стратегий и приемов моделирования ситуаций реального общения и организации взаимодействия обучающихся в группе (в парах, в малых группах) с целью совместного решения образовательных задач. В качестве традиционных приёмов данной технологии используется диалогическая, парная, групповая работа, нетрадиционных форм организации образовательной деятельности: игровые формы, техническая мастерская, «конструкторское бюро»;

- технология проектного обучения позволяет педагогу ориентировать обучающихся на самостоятельную поисковую, исследовательскую, рефлексивную, практическую, презентативную работу, результат которой имеет практический характер, важное прикладное значение, интересен и значим для обучающихся;

- кейсовая образовательная технология (Case Study) – это обучение действием: усвоение знаний и формирование умений есть результат активной самостоятельной деятельности обучающихся по разрешению противоречий, в результате чего и происходит творческое овладение профессиональными знаниями, навыками, умениями и развитие мыслительных способностей.

Формы контроля:

- беседа – вопросно-ответный метод контроля; применяется с целью активизации умственной деятельности обучающихся в процессе приобретения новых знаний или повторения и закрепления полученных ранее;

- наблюдение – педагог опосредованно контролирует выполнение того или иного задания обучающимися, при необходимости вносит коррективы;

- взаимоконтроль – обучающийся проверяет работу, выполненную другим обучающимся, по образцу, памятке или инструкции;

- творческие задания – учебные задания, для выполнения которых обучающийся должен применить нестандартное решение;

- технические задачи – проблемные ситуации в области конструирования, технического обслуживания того или иного объекта, предмета, разрешение которых связано с открытием и освоением нового познавательного действия.
- практическое задание - особый вид учебных занятий, имеющих целью практическое усвоение основных положений по предмету.

2.5. Оценочный материал

Оценочные материалы размещены в Приложении.

2.6. Список литературы и интернет – источников

Советов, Б. Я. Информационные технологии : учебник для среднего профессионального образования / Б. Я. Советов, В. В. Цехановский. — 8-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2024. — 414 с. — (Профессиональное образование). — ISBN 978-5-534-20053-9. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/557504> (дата обращения: 10.10.2024).

Хлебников, А. А., Информационные технологии : учебник / А. А. Хлебников. — Москва : КноРус, 2022. — 465 с. — ISBN 978-5-406-08923-1. — URL: <https://book.ru/book/942103> (дата обращения: 10.10.2024). — Текст : электронный.

**Контрольно-измерительные материалы для промежуточной аттестации
по дополнительной общеобразовательной программе
«Информационные технологии» нулевая линия**

Объединение: «IT-квантум»

Цель: выявление степени сформированности специальных компетенций обучающихся, прошедших курс обучения по дополнительной общеобразовательной (общеразвивающей) программе «Информационные технологии» нулевая линия

Задачи:

1. создать условия для представления обучающимися творческого(-их) продукта(-ов), созданных по итогам освоения программы;
2. определить уровень теоретической и практической подготовки учащихся;
3. определить уровень сформированности основных общеучебных компетенций;
4. проанализировать полноту реализации программы;
5. проанализировать актуальность содержания программы, при необходимости внести изменения, соответствующие уровню развития науки, техники, культуры, экономики, технологий и социальной сферы.

Формы аттестации: - тестирование;
- практическая работа;
- индивидуальный учебный проект;
- групповая проектная деятельность.

Критерии оценки результатов освоения программы по итогам выполнения тестирования и решения задач:

от 57-84 баллов - высокий уровень освоения программы
от 29-56 баллов - средний уровень освоения программы
от 0-28 баллов - низкий уровень освоения программы

Инструкция по выполнению:

Работа состоит из 20 заданий. Всего за тест можно набрать 54 балла и 15 баллов за практическую работу. Максимальное количество баллов за тест – 54 (1-18 задания), за практическую работу – 15 (19 задание), за презентацию индивидуального/группового проекта – 15 (20 задание).

Тестирование

Внимательно прочитайте каждое задание и предлагаемые варианты ответов. Отвечайте только после того, как вы поняли вопрос и проанализировали все варианты ответов.

Выполняйте задания в комфортном для Вас порядке. Отвечая на вопросы и выполняя задания, не спешите, так как ответы не всегда очевидны и требуют применения эрудиции.

Практическая работа

Для выполнения практического задания аттестующийся получает 3 случайных алгоритмических задачи на знание языковых конструкций Python. Задачи выбираются псевдослучайным образом из списка задач для выполнения 19 задания соответствующими инструментами (random.org и т.п.).

Решайте задачи в комфортном для Вас порядке. Под каждую задачу необходимо создать новый файл с расширением .py. Каждая задача проверяется по следующим критериям:

- Запуск программы без ошибок парсинга (программа содержит синтаксически корректный код без ошибок, возникающих на стадии чтения кода интерпретатором) – 1 балл;
- Устойчивость к ошибкам (программа не «падает» с ошибкой в процессе выполнения кода – отсутствие исключений) – 1 балл;
- Логически корректный код (логика программы соответствует описанию задачи) – 1 балл;
- Использование необходимых языковых конструкций (код программы использует языковые конструкции, которые соответствуют описанию/теме задачи) – 1 балл;
- Работоспособность программы при различных исходных данных (код программы должен уметь обрабатывать разные исходные данные) – 1 балл.

Проектная деятельность

Аттестующийся/группа аттестующихся выбирает одну из проектных тем, изученных в процессе прохождения курса «Информационные технологии» (сетевой интерфейс socket, база данных SQLite, Telegram Bot, графическая библиотека Tkinter), и создает проект, который необходимо презентовать перед аттестационной комиссией.

Критерии оценки проекта:

- высокий уровень освоения – от 11 до 15 баллов
- средний уровень освоения – от 6 до 10 баллов
- низкий уровень освоения – от 0 до 5 баллов

Оценка определяет уровень освоения проектных тем в рамках учебной программы «Информационные технологии».

Баллы, полученные Вами за выполненные задания, суммируются. Постарайтесь выполнить как можно больше заданий и набрать наибольшее количество баллов.

Желаем успеха!

Тестирование

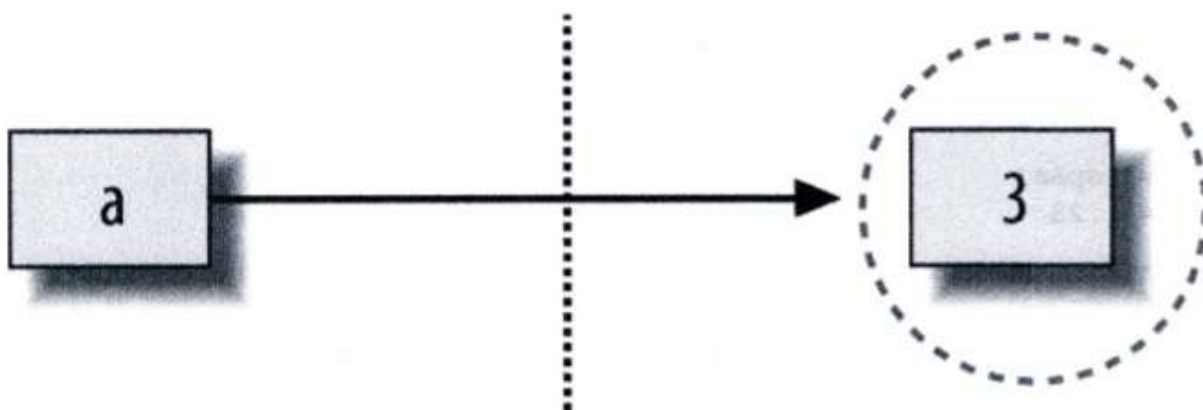
1. Установите соответствие между объектом и типом данных, к которому относится этот объект: к каждой позиции, данной в первом столбце, подберите соответствующую позицию из второго столбца.

<u>ТИП ДАННЫХ</u>	<u>ОБЪЕКТ</u>
А) dict	1) 125.0
Б) int	2) {1, 2, 3, 4, 5}
В) float	3) ['молоко', 'сыр', 'творог', 'кефир']
Г) set	4) {'Михаил': '17 лет', 'Ирина': '20 лет', 'Ян': '13 лет'}
Д) list	5) 125

Запишите в ответ цифры, расположив их в порядке, соответствующем буквам:

А	Б	В	Г	Д

2. Все ли утверждения верны? Если нужно, укажите неверные утверждения.



- 1) Переменная в Python – это символическое имя, которое ссылается (указывает) на объект.
- 2) Объекты — это области выделенной памяти с достаточным пространством для представления значений, для которых они предназначены. Каждый объект содержит значение, тип и счетчик ссылок.

- 3) Переменные и объекты хранятся в разных местах памяти и связываются посредством ссылок.
 - 4) Ссылки — это следуемые указатели от переменных к объектам.
 - 5) Когда мы создаём объект и присваиваем его переменной, переменная только ссылается на объект, а не представляет собой этот объект.
 - 6) Когда переменная встречается в выражении, она тотчас же заменяется объектом, на который в текущий момент ссылается.
3. Установите соответствие между терминами и определениями: к каждой позиции, данной в первом столбце, подберите соответствующую позицию из второго столбца.

Термины	Определения
А) Логическая строка	1) Пробелы и табуляции, которые используются для группировки предложений
Б) Точка с запятой (;)	2) Это то, что вы видите, когда набираете программу
В) Отступы	3) Набор предложений, которые имеют одинаковый отступ
Г) Физическая строка	4) Символ, который отмечает конец логической строки
Д) Блок	5) Это то, что Python видит как единое предложение

Запишите в ответ цифры, расположив их в порядке, соответствующем буквам:

А	Б	В	Г	Д

4. Установите соответствие между терминами и определениями: к каждой позиции, данной в первом столбце, подберите соответствующую позицию из второго столбца.

Термины	Определения
А) Операторы	1) Некий функционал, производящий какие-либо действия, который может быть представлен в виде символов

	или специальных зарезервированных слов
Б) Ассоциативность	2) Свойство оператора/операции, влияющее на очередность его выполнения в выражении
В) Операнды	3) Это означает, что операторы с равным приоритетом будут обработаны по порядку от левого до правого. Например, $2 + 3 + 4$ обрабатывается как $(2 + 3) + 4$
Г) Приоритет	4) Данные, с которыми производятся операции при выполнении программы

Запишите в ответ цифры, расположив их в порядке, соответствующем буквам:

А	Б	В	Г

5. Дано выражение: $(1 + 3) * 6 == 10 + 2 * 7$

В 1 строчку через запятую выпишите операторы, а во 2 – операнды, которые встречаются в выражении. А также укажите результат, который получится при выполнении данного выражения.

Операторы	
Операнды	
Результат	

6. Укажите варианты ответов, в которых дано верное условие, необходимое для корректной работы программы.

```

1  # password - пароль
2  # attempts - попытки
3  # guess - ввод пользователя
4
5  password = 'qewerty'
6  attempts = 3
7
8  while УСЛОВИЕ:
9      guess = input('Введите пароль: ')
10
11     if guess == password:
12         print('Добро пожаловать!')
13
14         break
15
16     else:
17         attempts -= 1
18         print('Неверный пароль. Осталось попыток:', attempts)
19
20     else:
21         print('У вас закончились попытки, аккаунт заблокирован!')

```

Если ответов несколько запишите их в порядке возрастания через запятую.

1) attempts == 0	2) attempts > 0
3) attempts != 0	4) attempts < 0
Ответ:	

7. Установите соответствие между терминами и фрагментами кода: к каждой позиции, данной в первом столбце, подберите соответствующую позицию из второго столбца.

```

1  def get_max(a, b):
2      '''Наибольшее число.
3
4      Функция принимает 2 числа.
5      Функция сравнивает числа и возвращает наибольшее.
6      В случае, если числа равны, возвращает "Числа равны".'''
7
8      if a > b:
9          return a
10
11     elif a < b:
12         return b
13
14     else:
15         return "Числа равны"
16
17     print(get_max.__doc__)
18     print(get_max(3, b=4))

```

Термины	Фрагменты кода
А) Параметры функции	1) return
Б) Позиционный аргумент	2) b=4
В) Ключевой аргумент	3) __doc__
Г) Оператор возврата и выхода из функции	4) a, b
Д) Атрибут, предназначенный для хранения строки документации	5) 3

Запишите в ответ цифры, расположив их в порядке, соответствующем буквам:

А	Б	В	Г	Д

8. Укажите вариант ответа, в котором даны числа, напечатанные в результате выполнения программы.

```

1      x = 1
2
3      def func():
4          # global x
5
6          x = 2
7          print(x)
8
9      func()
10     print(x)

```

1) 1, 1	2) 1, 2
3) 2, 2	4) 2, 1
Ответ:	

9. Установите соответствие между объектом и типом данных, к которому относится этот объект: к каждой позиции, данной в первом столбце, подберите соответствующую позицию из второго столбца.

ОБЪЕКТЫ

ТИП ДАННЫХ

А) ['C++', 'Java', 'C#', 'PHP', 'JavaScript', 'Python']

1) Изменяемый тип

Б) 1538.9

2) Неизменяемые тип

В) {'Марка': 'Toyota', 'Модель': 'Premio', 'Год': 2012}

Г) 'Это строка'

Д) None

Е) frozenset({0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89})

Ё) True

Ж) (2, 3, 11, 17, 131, 523)

Запишите в ответ цифры, расположив их в порядке, соответствующем буквам:

А	Б	В	Г	Д	Е	Ё	Ж

10. Укажите вариант ответа, в котором дан список, напечатанный в результате выполнения программы.

```
1 array = list()
2 for number in range(1, 10):
3     if number % 2 == 0:
4         array.append(number)
5
6     else:
7         del array[-1]
8
9 print(array)
```

1) [2, 4, 6]	2) [1, 3, 5, 7, 9]
3) [2, 4, 6, 8, 10]	4) [2, 4, 6, 8]
Ответ:	

11. Укажите вариант ответа, в котором дано количество пар «ключ-значение» в словаре, напечатанное в результате выполнения программы.

```
1 address_book = {'Яков': 'jacob2004@gmail.com',
2                 'Мария': 'masha2001@mail.ru',
3                 'Михаил': 'misha2007@icloud.com',
4                 'Анастасия': 'nastya1999@inbox.ru'}
5
6 del address_book['Михаил']
7
8 address_book['Ксения'] = 'ksusha2005@gmail.com'
9
10 address_book['Анастасия'] = 'anastasia1999@gmail.com'
11
12 length = len(address_book)
13
14 print('Количество контактов в адресной книге:', length)
```

1) 3	2) 5
3) 4	4) 0
Ответ:	

12. Верно ли утверждение о том, что **множества** – это неупорядоченные наборы неизменяемых объектов?

13. Укажите вариант ответа, в котором дано множество, напечатанное в результате выполнения программы.

```
1 array = [4, 1, 4, 3, 3, 2, 3, 4, 1, 5]
2 array = set(array)
3 print(array)
```

1) {4, 1, 4, 3, 3, 2, 3, 4, 1, 5}	2) {1, 1, 2, 3, 3, 3, 4, 4, 4, 5}
3) {4, 1, 3, 2, 5}	4) {1, 2, 3, 4, 5}
Ответ:	

14. Дан фрагмент кода. Возникнет ли ошибка при запуске такой программы? Если возникнет, то почему?

```
1 array = {1, '343', [4, 5, 6], ('7', '8', '9')}
2 print(array)
```

15. Дан фрагмент кода. Возникнет ли ошибка при запуске такой программы? Если возникнет, то почему?

```
1 text = 'Hello, world!'
2 del text[-1]
3 print(text)
```

16. На рабочем столе размещен текстовый файл «Текстовый документ». Программа, исходный код которой представлен ниже, должна открыть файл, прочитать его содержимое и закрыть файл. Но при запуске написанной программы возникает сообщение об ошибке. Найдите и исправьте ошибку в коде.

```
1 with open(file=r'C:\Users\George\Desktop\Текстовый документ.txt', mode='w') as f:
2     text = f.read()
3     print(text)
```

17. Верно ли утверждение о том, что else-блок выполняется в любом случае – даже если в try-блоке произошла ошибка? Если утверждение неверно, то какую инструкцию необходимо вставить вместо else?

```
1  try:
2      result = 1 / 0
3      print('Ответ:', result)
4
5  except:
6      print('Делить на ноль нельзя!')
7
8  else:
9      print('Программа завершена.')
```

18. Установите соответствие между терминами и фрагментами кода: к каждой позиции, данной в первом столбце, подберите соответствующую позицию из второго столбца.

```

3 class SchoolMember(metaclass=abc.ABCMeta):
4     amount = 0
5
6     def __init__(self, name, age):
7         SchoolMember.amount += 1
8         self.name = name
9         self.age = age
10
11     @abc.abstractmethod
12     def tell(self):
13         print('Имя: {}, возраст: {}'.format(self.name, self.age), end=' ')
14
15 class Teacher(SchoolMember):
16     amount = 0
17
18     def __init__(self, name, age, salary):
19         Teacher.amount += 1
20         SchoolMember.__init__(self, name, age)
21         self.salary = salary
22
23     def tell(self):
24         SchoolMember.tell(self)
25         print('зарплата: {}'.format(self.salary))

```

Термины	Фрагменты кода
А) Декоратор, объявляющий метод класса абстрактным	1) class SchoolMember
Б) Встроенный метакласс, объявляющий класс абстрактным базовым классом	2) abc.ABCMeta
В) Специальный метод инициализации (конструктор)	3) amount
Г) Параметр, ссылающийся на объект, через который был вызван метод	4) def __init__()
Д) Поле объекта	5) self
Е) Подкласс	6) self.name
Ё) Поле класса	7) @abc. abstractmethod

Ж) Базовый (родительский) класс	8) def tell()
З) Метод класса	9) class Teacher(SchoolMember)

Запишите в ответ цифры, расположив их в порядке, соответствующем буквам:

А	Б	В	Г	Д	Е	Ё	Ж	З

19. Практическая работа.

Аттестуемому необходимо выбрать любые 3 темы, изученных в рамках курса «Информационные технологии», и обратиться к лицу (педагогу), принимающему аттестацию, для получения заданий по выбранным темам.

Практическая работа проверяется и оценивается лицом, проводящим аттестацию. Список тем для выполнения практического задания находится ниже.

Список тем:

- Действия с числами
- Преобразование символьных величин
- Работа со списками
- Множества
- Словари
- Файлы

Список задач для выполнения 19 задания:

- Действия с числами
 1. Составить программу, проверяющую, будет ли простым данное натуральное число.
 2. Написать программу, которая выводит на экран первые N простых чисел.
 3. Найти все делители натурального числа N.

4. Натуральное число называется совершенным, если оно равно сумме всех своих делителей, включая единицу. Вывести первые N ($N < 5$) совершенных чисел на экран.
5. Найти все целые числа из интервала от N до M , которые делятся на каждую из своих цифр.
6. Найти все целые числа из интервала от N до M , которые делятся на сумму всех своих цифр.
7. Проверить, являются ли два числа дружественными (сумма делителей одного числа равна другому числу). Например, сумма делителей числа 220 равна: $1 + 2 + 4 + 5 + 10 + 11 + 20 + 22 + 44 + 55 + 110 = 284$, а сумма делителей числа 284 равна: $1 + 2 + 4 + 71 + 142 = 220$, поэтому числа 220 и 284 - дружественные.
8. Определим операцию $\#$ так, что $A \# B = A - B + A \% B$. Найти все числа из интервала от N до M , для которых эта операция коммутативна.
9. Характером натурального числа назовем сумму всех его делителей, не равных единице и самому числу. Характером простого числа будем считать нуль. Написать программу, которая вычисляет характер числа.
10. Число из n цифр называется числом Армстронга, если сумма его цифр в степени n равна самому числу. Например: $1634 = 1^4 + 6^4 + 3^4 + 4^4$. Составить программу, находящую все числа Армстронга из двух и трех цифр.
11. Составить программу, находящую все трехзначные числа abc такие, что $abc = a! + b! + c!$.

- Преобразование символьных величин

1. Дана строка символов. Сформировать новую строку, состоящую из символов с номерами три, шесть, девять и т.д. данной строки.
2. Даны две строки, Вывести большую по длине строку столько раз, на сколько символов отличаются строки.
3. Дана строка символов. Определите, какой из символов «s» или «q» встречается в ней раньше. Если какой-то из символов в строке отсутствует, то сообщить об этом.
4. Написать программу, которая считает число слов в предложении.
5. Написать программу, которая проверяет, можно ли из букв, входящих в слово A , составить слово B .

6. Написать программу, которая удаляет в данном тексте все лишние пробелы.
7. Перепечатать заданный текст, удалив из него знаки "+", непосредственно за которыми идет цифра.
8. Перепечатать заданный текст, удалив из него все буквы b, непосредственно перед которыми находится буква c.
9. Дана строка, состоящая из слов, разделенных символами, которые перечислены во второй строке. Показать все слова.

- Работа со списками

1. Найдите модуль разности первого и последнего элементов числового списка.
2. Определите индекс элемента числового списка, значение которого равно сумме первого и последнего элемента того же списка.
3. Определите, равен ли "центральный" элемент числового списка произведению крайних элементов.
4. Определите, является ли произведение элементов числового списка факториалом числа, равного длине списка.
5. Определите, все ли элементы числового списка являются трёхзначными числами.
6. Определите, все ли элементы числового списка являются чётными числами.
7. По заданному списку целых чисел создайте список, состоящий из их остатков от деления на 2.
8. Определите, встречается ли указанный своим индексом элемент первого заданного списка во втором заданном списке.
9. Составьте программу, которая по двум спискам, имеющим одинаковую длину, возвращает:
 - a. первый список, если сумма элементов первого списка совпадает с суммой элементов второго списка
 - b. сумму элементов второго списка в противном случае
10. Напишите программу, определяющую в списке, содержащем списки целых чисел, количество таких списков, содержащих заданное число.
11. Напишите программу, удаляющую «крайние» элементы списка.

12. Напишите программу, меняющую местами «крайние» элементы списка.
13. Напишите программу, меняющую местами пару элементов списка по их указанным индексам.
14. Напишите программу, добавляющую в начало списка копию его предпоследнего элемента.
15. Напишите программу, удваивающую первый элемент списка (путём создания копии), если его значение меньше значения второго элемента.
16. Напишите программу, объединяющую первую «половину» элементов первого списка со второй «половиной» элементов второго списка.
17. Напишите программу, объединяющую три заданных списка в один, после предварительного удаления из них последних элементов.
18. Напишите программу, конструирующую список из «центральных» элементов трёх заданных списков.

- Множества

1. Даны 2 множества. Найти:
 - a. объединение множеств
 - b. пересечение множеств
 - c. разность множеств
 - d. множество из элементов, встречающихся в одном множестве, но не встречающихся в обоих

- Словари

1. Опишите, используя словарь, телефонную книгу. Составьте программу, выдающую список абонентов, имеющих телефонный номер, начинающийся на 33.
2. Опишите, используя словарь, школьную нагрузку (фамилия преподавателя, класс, часы). Составьте программу, определяющую нагрузку каждого преподавателя. Определить, у какого преподавателя самая большая нагрузка и у кого самая низкая.
3. Опишите, используя словарь, таблицу соревнований (название команды, количество набранных очков). Выбрать команду, занявшую первое место. Упорядочить список команд, в зависимости от занятого места.

4. При сдаче норм ГТО были получены результаты забега на 100 метров и прыжков в длину. Задайте нормы ГТО по этим видам, определите списки учеников, не выполнивших нормативы, количество учеников, сдавших нормативы, а также список 3-х лучших.
5. Опишите, используя словарь, товар (наименование товара, старая цена, новая цена). Составьте программу, определяющую, на какие товары повысятся цены и на сколько.
6. В анкетных данных обозначены фамилия, пол, рост. Определите средний рост женщин, фамилию самого высокого мужчины, есть ли в группе хотя бы два человека одного роста.

- Файлы

1. Дан файл f , компоненты которого являются действительными числами. Найти:
 - a. сумму компонент файла f
 - b. произведение компонент файла f
 - c. сумму квадратов компонент файла f
 - d. модуль суммы и квадрат произведения компонент файла f
 - e. последнюю компоненту файла
2. Дан файл f , компоненты которого являются действительными числами. Найти:
 - a. наибольшее из значений компонент
 - b. наименьшее из значений компонент с четными номерами
 - c. наибольшее из значений модулей компонент с нечетными номерами
 - d. разность первой и последней компонент файла
3. Дан файл f , компоненты которого являются целыми числами. Найти:
 - a. количество четных чисел среди компонент
 - b. количество удвоенных нечетных чисел среди компонент
 - c. количество квадратов нечетных чисел среди компонент

4. Даны файлы f_1, f_2, f_3, f_4, f_5 , компоненты которых являются действительными числами. Организовать обмен компонентами между файлами в соответствии со следующей схемой:
 $f_1 \rightarrow f_3, f_2 \rightarrow f_4, f_3 \rightarrow f_5, f_4 \rightarrow f_2, f_5 \rightarrow f_1$, т.е. компоненты файла f_1 переписываются в файл f_3 , компоненты файла f_2 - в f_4 и т.д.
Разрешается использовать только один вспомогательный файл h .
5. Дан символьный файл f . В файле не менее двух компонент. Определить, являются ли два первых символа файла цифрами. Если да, то установить, является ли число, образованное этими цифрами, четным.
6. Дан файл f , компоненты которого являются целыми числами. Получить в файле g все компоненты файла f :
 - a. являющимися четными числами
 - b. делящиеся на 3 и не делящиеся на 7
 - c. являющимися точными квадратами
7. Дан символьный файл f . Получить файл g , образованный из файла f заменой всех его прописных (больших) букв одноименными строчными (малыми).
8. Дан файл f , компоненты которого являются целыми числами. Записать в файл g все четные числа файла f , а в файл h - все нечетные. Порядок следования чисел сохраняется.
9. Дан символьный файл f . Записать в файл g компоненты файла f в обратном порядке.
10. Дан файл f , компоненты которого являются целыми числами. Получить файл g , образованный из файла f исключением повторных вхождений одного и того же числа.
11. Дан файл f , компоненты которого являются целыми числами. Никакая из компонент файла не равна нулю. Файл f содержит столько же отрицательных чисел, сколько и положительных. Используя вспомогательный файл h , переписать компоненты файла f в файл g так, чтобы в файле g :
 - a. не было двух соседних чисел с одним знаком
 - b. сначала шли положительные, потом отрицательные числа

- с. числа шли в следующем порядке: два положительных, два отрицательных, два положительных, два отрицательных и т.д. (предполагается, что число компонент в файле f делится на 4)

12. Дан символьный файл f , содержащий сведения о сотрудниках учреждения, записанные по следующему образцу:
 фамилия_имя_отчество, фамилия_имя_отчество, ... Записать эти сведения в файле g , используя образцы:

- а. имя_отчество_фамилия, имя_отчество_фамилия...
 б. фамилия_и.о., фамилия_и.о...

Ключ ответов

Номер задания	Ответ	Максимальное количество баллов
1	45123	5
2	Все утверждения верны	1
3	54123	5
4	1342	4
5	+, *, == 1, 3, 6, 10, 2, 7 True	3
6	2, 3	2
7	45213	5
8	4	1
9	12122222	8
10	1	1
11	3	1
12	Верно	1

13	4	1
14	Ошибка возникает. Элементом множества не может являться изменяемый объект - список	2
15	Ошибка возникает. Строка - неизменяемый объект, который не поддерживает операцию удаления элемента	2
16	Файл необходимо открыть в режиме чтения 'r', а не в режиме записи 'w'	1
17	Утверждение неверно. finally	2
18	724569318	9
19		15 (3 задачи × 5 баллов)
20		15