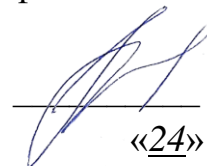


Федеральное государственное образовательное бюджетное
учреждение высшего образования
«Финансовый университет при Правительстве Российской Федерации»
(Финансовый университет)
Липецкий филиал Финуниверситета

УТВЕРЖДАЮ
Заместитель директора
по учебно-методической работе
Липецкого филиала Финуниверситета


О.Н. Левчegov
«24» апреля 2024 г.

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ
ДИСЦИПЛИНЫ «ОП.05 ОСНОВЫ АЛГОРИТМИЗАЦИИ И
ПРОГРАММИРОВАНИЯ»

по специальности 10.02.04 Обеспечение информационной безопасности
телекоммуникационных систем

Липецк - 2024

Фонд оценочных средств разработан на основе федерального государственного образовательного стандарта среднего профессионального образования (далее – ФГОС СПО) по специальности 10.02.04 «Обеспечение информационной безопасности телекоммуникационных систем».

Разработчики:

Якушов Юрий Алексеевич старший преподаватель кафедры Учет и информационные технологии в бизнесе Липецкого филиала Финуниверситета.

Фонд оценочных средств рассмотрен и рекомендован к утверждению на заседании кафедры Учет и информационные технологии в бизнесе Липецкого филиала Финуниверситета.

Протокол от 23.04.2024 г. №10

Заведующий кафедрой

Учет и информационные технологии в бизнесе _____ Н.С. Морозова



1. ОБЩИЕ ПОЛОЖЕНИЯ

Фонды оценочных средств (далее ФОС) предназначены для контроля и оценки образовательных достижений обучающихся, освоивших программу учебной дисциплины «ОП.05 Основы алгоритмизации и программирования».

ФОС включают контрольные материалы для проведения текущего контроля и итоговой аттестации в форме экзамена.

ФОС разработаны на основании положений:

- ФГОС СПО) по специальности 10.02.04 Обеспечение информационной безопасности телекоммуникационных систем, утвержденного приказом Министерства образования и науки от 9 декабря 2016 №1551;
- профессиональных стандартов:
 - Приказ Минтруда России № 536н от 14 сентября 2022 г. Об утверждении профессионального стандарта «Специалист по защите информации в телекоммуникационных системах и сетях»;
 - Указ Президента Российской Федерации от 7 мая 2024 г. N 309 "О национальных целях развития Российской Федерации на период до 2030 года и на перспективу до 2036 года";
- Национальная программа «Цифровая экономика Российской Федерации», утвержденная протоколом заседания президиума Совета при Президенте Российской Федерации по стратегическому развитию и национальным проектам от 4 июня 2019 г. № 7;
- Постановление Правительства Российской Федерации от 18.04.2016 г. № 317 «О реализации национальной технологической инициативы»;
- Проект «Современная цифровая образовательная среда в Российской Федерации» (утверждён президиумом Совета при Президенте Российской Федерации по стратегическому развитию и приоритетным проектам, протокол от 25.10.2016 № 9);
- Распоряжение Правительства РФ от 20.02.21 № 431-р «Концепция цифровой и функциональной трансформации социальной сферы, относящейся к сфере деятельности Министерства труда и социальной защиты Российской Федерации, на период до 2025 года».

2.РЕЗУЛЬТАТЫ ОСВОЕНИЯ ДИСЦИПЛИНЫ, ПОДЛЕЖАЩИЕ ПРОВЕРКЕ

В рамках программы дисциплины обучающимися осваиваются следующие компетенции:

Общие компетенции

Код компетенции	Формулировка компетенции	Знания, умения
ОК 01	<p>Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам</p> <p>(Ключевые компетенции цифровой экономики</p> <p>Креативное мышление</p> <p>Критическое мышление в цифровой среде)</p>	<p>Умения:</p> <ul style="list-style-type: none"> – распознавать задачу и/или проблему в профессиональном и/или социальном контексте; – анализировать задачу и/или проблему и выделять её составные части; – определять этапы решения задачи; выявлять и эффективно искать информацию, необходимую для решения задачи и/или проблемы; – составить план действия; определить необходимые ресурсы; – владеть актуальными методами работы в профессиональной и смежных сферах; – реализовать составленный план; – оценивать результат и последствия своих действий (самостоятельно или с помощью наставника) <p>использовать цифровые средства и ресурсы для генерирования новых идей и решений;</p> <ul style="list-style-type: none"> - абстрагироваться от стандартных моделей: перестраивать сложившиеся способы решения задач, выдвигать альтернативные варианты действий с целью выработки новых оптимальных алгоритмов действий; - использовать цифровые средства и приложения для создания продукта. – формировать и проверять гипотезы; - выбирать и использовать уместные цифровые средства, приложения и ресурсы для постановки и решения задачи/проблемы; - оценить информацию/данные на достоверность и релевантность сравнением нескольких источников информации; - разделять комплексные задачи на подзадачи; отслеживать процесс исполнения задач помощью цифровых инструментов - строить логические умозаключения на основании информации/данных, в том числе в различных цифровых средах (в том числе, оценивать результат и последствия своих действий).

	<p>(Ключевые компетенции цифровой экономики</p> <p>Креативное мышление</p> <p>Критическое мышление в цифровой среде)</p>	<p>Знания:</p> <ul style="list-style-type: none"> – актуальный профессиональный и социальный контекст, в котором приходится работать и жить; – основные источники информации и ресурсы для решения задач и проблем в профессиональном и/или социальном контексте. – алгоритмы выполнения работ в профессиональной и смежных областях; – методы работы в профессиональной и смежных сферах; – структуру плана для решения задач; – порядок оценки результатов решения задач профессиональной деятельности. <p>-возможностей и ограничений цифровой среды и цифровых инструментов для создания продукта/решения задачи;</p> <p>- цифровых инструментов для разработки и создания продукта;</p> <p>- принципов работы социальных сетей и медиа с точки зрения создания оригинального продукта (понимание трендов, предпочтений пользователей).</p> <p>-цифровые инструменты и сервисы для проверки достоверности информации/гипотезы;</p> <p>- методы и приемы формулирования гипотез и задач;</p> <p>-цифровые ресурсы для решения задач/проблем в профессиональном и/или социальном контексте и для оценки результатов решения</p>
ОК 02	<p>Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности</p> <p>(Ключевые компетенции цифровой экономики</p> <p>Управление информацией и данными</p>	<p>Умения:</p> <ul style="list-style-type: none"> – определять задачи для поиска информации; – определять необходимые источники информации; – планировать процесс поиска; структурировать получаемую информацию; – выделять наиболее значимое в перечне информации; – оценивать практическую значимость результатов поиска; – оформлять результаты поиска. <p>-выбирать оптимальный формат, способ и место хранения информации и данных с помощью цифровых инструментов;</p> <p>- защитить информацию (данные) при помощи паролей и кодирования;</p> <p>- создавать резервные копии данных на различных носителях;</p> <p>- искать информацию в сети Интернет с использованием фильтров и ключевых слов;</p> <p>- оценивать данные на достоверность;</p> <p>-идентифицировать различные виды мошенничества с персональными данными;</p> <p>-оформлять результаты поиска с помощью цифровых инструментов.</p> <p>-применять программные решения для структурирования и систематизации информации;</p> <p>- оценить информацию/данные на достоверность и релевантность сравнением нескольких источников</p>

	<p>Критическое мышление в цифровой среде)</p> <p>(Ключевые компетенции цифровой экономики Управление информацией и данными</p> <p>Критическое мышление в цифровой среде)</p>	<p>информации; - оценивать практическую значимость результатов поиска с помощью цифровых инструментов.</p> <p>Знания: – источников, применяемых в профессиональной деятельности; – приемы структурирования информации; – формат оформления результатов поиска информации - инструменты крупнейших цифровых экосистем для получения, обработки и анализа информации; - особенности различных расширений и форматов хранения данных; - принципы работы различных поисковых сервисов; - риски публикации персональных данных и их отображения в социальных сетях; - нормы интеллектуальной собственности, лицензий и др. норм при публикации и скачивании контента. - способы и цифровые инструменты/ сервисы для проверки достоверности информации.</p>
ОК 03	<p>Планировать и реализовывать собственное профессиональное и личностное развитие.</p> <p>(Ключевые компетенции цифровой экономики Саморазвитие в условиях неопределенности)</p>	<p>Умения: определять актуальность нормативно-правовой документации в профессиональной деятельности; применять современную научную профессиональную терминологию; определять и выстраивать траектории профессионального развития и самообразования. -- ставить себе образовательные цели под возникающие жизненные задачи; –находить информацию в целях самообразования и обучения, - создавать электронные конспекты при помощи онлайн платформ для создания, представления и анализа презентаций; – самостоятельно определять пробелы в своих знаниях и компетенциях с использованием инструментов самооценки и цифровых оценочных средств LMS, – выбирать цифровые средства в целях саморазвития, использовать цифровые тренажеры для обучения программированию - адаптироваться к появлению новых цифровых средств, приложений, программных обеспечений</p>

	(Ключевые компетенции цифровой экономики Саморазвитие в условиях неопределенности)	Знания: - содержание актуальной нормативно-правовой документации; - современная научная и профессиональная терминология; - возможные траектории профессионального развития. -основных образовательных Интернет-ресурсов, типов цифрового образовательного контента; - возможностей и ограничений образовательного процесса при использовании цифровых технологий. - возможности, область применения и интерфейс цифровых инструментов для обучения
ОК 04	Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами. (Ключевые компетенции цифровой экономики Коммуникация и кооперация в цифровой среде) (Ключевые компетенции цифровой экономики Коммуникация и кооперация в цифровой среде)	Умения: определять актуальность нормативно-правовой документации в профессиональной деятельности; применять современную научную профессиональную терминологию; определять и выстраивать траектории профессионального развития и самообразования. – ставить себе образовательные цели под возникающие жизненные задачи; - находить информацию в целях самообразования и обучения при помощи цифровых инструментов: Знания: -психологические основы деятельности коллектива, психологические особенности личности; -основы проектной деятельности. - самостоятельно определять пробелы в своих знаниях и компетенциях с использованием инструментов самооценки и цифровых оценочных средств; - выбирать цифровые средства в целях саморазвития; адаптироваться к появлению новых цифровых средств, приложений, программных обеспечений.
ОК 05	Осуществлять устную и письменную коммуникацию на государственном языке с учетом особенностей социального и культурного контекста. (Ключевые компетенции цифровой экономики)	Умения: – грамотно излагать свои мысли и оформлять документы по профессиональной тематике на государственном языке, – проявлять толерантность в рабочем коллективе. -выбирать цифровые средства общения в соответствии с целью взаимодействия и индивидуальными особенностями (в том числе культурными) собеседника; - использовать цифровые средства общения при взаимодействии с другими людьми, в том числе для организации совместной деятельности; - справляться с нежелательным поведением других людей в цифровой среде (угрозы, травля, агрессивные действия);

	Коммуникация и кооперация в цифровой среде)	<ul style="list-style-type: none"> - выбирать цифровые медиа (текст, фото, видео, анимация и т.п.) в соответствии с культурными, познавательными и личностными особенностями собеседника; - находить тематические Интернет-сообщества
	(Ключевые компетенции цифровой экономики Коммуникация и кооперация в цифровой среде)	<p>Знания:</p> <ul style="list-style-type: none"> - особенности социального и культурного контекста; - правила оформления документов и построения устных сообщений. - видов и функций информационных сообщений, групп информационных объектов; -каналов распространения информации и организации совместной работы (командной работы); - преимуществ и ограничений цифровых средств при общении и совместной работе; - культуру общения, принятую в цифровой среде; - принципы создания и функционирования Интернет-сообществ.
ОК 09	Использовать информационные технологии в профессиональной деятельности (Ключевые компетенции цифровой экономики Элемент цифровой грамотности в профессиональной деятельности)	<p>Умения:</p> <ul style="list-style-type: none"> – применять средства информационных технологий для решения профессиональных задач; – использовать современное программное обеспечение. – использовать интегрированные среды программирования для написания, трансляции и исполнения программного кода <p>Знания:</p> <ul style="list-style-type: none"> – современные средства и устройства информатизации; порядок их применения и программное обеспечение в профессиональной деятельности – возможности, область применения и интерфейс интегрированных сред программирования, интегрированную среду разработки для совместной работы в браузере
ОК 10	Пользоваться профессиональной документацией на государственном и иностранном языках.	<p>Умения:</p> <ul style="list-style-type: none"> – понимать общий смысл четко произнесенных высказываний на известные темы (профессиональные и бытовые), – понимать тексты на базовые профессиональные темы; – участвовать в диалогах на знакомые общие и профессиональные темы; – строить простые высказывания о себе и о своей профессиональной деятельности; – кратко обосновывать и объяснить свои действия (текущие и планируемые); – писать простые связные сообщения на знакомые или интересующие профессиональные темы

		<p>Знания:</p> <p>правила построения простых и сложных предложений на профессиональные темы;</p> <p>основные общеупотребительные глаголы (бытовая и профессиональная лексика);</p> <p>лексический минимум, относящийся к описанию предметов, средств и процессов профессиональной деятельности;</p> <p>особенности произношения;</p> <p>правила чтения текстов профессиональной направленности</p>
--	--	--

Профессиональные компетенции

Основные виды деятельности	Код и формулировка компетенции	Показатели освоения компетенции
Эксплуатация информационно-телекоммуникационных систем и сетей:	ПК 1.1. Производить монтаж, настройку, проверку функционирования и конфигурирование оборудования информационно-телекоммуникационных систем и сетей.	<p>Практический опыт:</p> <p>Разрабатывать алгоритм решения поставленной задачи и реализовывать его средствами автоматизированного проектирования.</p> <p>Владеть навыками использования инструментальных средств обработки информации сквозных цифровых технологий</p>
		<p>Умения:</p> <p>Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием.</p> <p>Оформлять документацию на программные средства.</p> <p>Использовать алгоритмы обработки информации для различных приложений сквозных цифровых технологий</p>
		<p>Знания:</p> <p>Основные этапы разработки программного обеспечения.</p> <p>Основные принципы технологии структурного и объектно-ориентированного программирования. Платформенные и сквозные цифровые технологии для создания, исполнения и управления информационной системой</p>
	ПК 1.4. Осуществлять контроль функционирования информационно-	<p>Практический опыт:</p> <p>Разрабатывать код программного продукта на языке высокого уровня Python в интегрированной среде программирования PyScripter на основе готовой спецификации на уровне модуля.</p>

	<p>телекоммуникационных систем и сетей</p>	<p>Разрабатывать блок-схемы с помощью инструментальных средств онлайн -сервисов Работать в команде разработчиков с использованием инструментальных онлайн средств Разрабатывать диалоговых чат-ботов, планировщики заданий и чат-ботов для Телеграмм</p> <p>Умения: Создавать программу по разработанному алгоритму как отдельный модуль. Оформлять документацию на программные средства. Создавать программы на языке высокого уровня python в интегрированных средах программирования с использованием онлайн – сервисов для совместной работы в браузере Использовать алгоритмы сравнения текстов, векторизацию, обучение машинной модели, измерять качества классификатора и интегрировать алгоритмы в чат-бот. Уметь применять инструментальные средства сервисов для создания блок-схем и визуальной совместной деятельности Уметь использовать возможности диалоговой платформ и фреймворков для автоматизации создания ботов</p> <p>Знания: Основные этапы разработки программного обеспечения. Основные принципы технологии структурного и объектно-ориентированного программирования. Алгоритмы сравнения текстов, векторизация, обучение машинной модели, измерение качества классификатора и интеграцию в чатбота. Назначение и возможности сервисов для создания блок-схем, Назначение и возможности пакета визуальных средств для совместной работы Назначение и возможности диалоговых платформ, фреймворков Синтаксис и семантика высокоуровневого языка python. Интегрированных сред программирования: PyScripter, Wing IDE, PyCharm, интегрированную среду разработки для совместной работы в браузере repl.it(https://replit.com)</p>
--	--	--

Объекты оценивания	Показатели оценки	Критерии оценки	Компетенции
У-1 Разрабатывать алгоритмы для конкретных задач.	Построение математической модели задачи, блок-схемы, программного кода	Математическая модель и блок-схема построены правильно, программный код соответствует правилам хорошего стиля программирования, выполнен без ошибок, соответствует требованиям МУ к ПР1.1-4.14	ОК 01, 02, 04, 05,09,10, ПК 1.1-1.5, ПК 2.4, 2.5
У-2 Использовать языки программирования	Использование языка программирования высокого уровня для разработки логически правильных и эффективных программ.	Программный код соответствует правилам хорошего стиля программирования и требованиям, определенным в МУ к МУ к ПР1.1-4.14	ОК 01, 02, 04, 05,09,10, ПК 1.1-1.5, ПК 2.4, 2.5
У-3 Использовать платформы, онлайн сервисы и конструкторы сквозных цифровых технологий	Использование алгоритмов обработки информации для различных приложений сквозных цифровых технологий при создании планировщика задач, чат-бота	Планировщик задач и чат-бот соответствует требованиям, определенным в МУ к П 2.9,3.4	ОК 01, 02, 04, 05,09,10, ПК 1.1-1.5, ПК 2.4, 2.5
У-4 Использовать программы для графического отображения алгоритмов.	Использование инструментальных средств сервиса draw.io (https://draw-io.softonic.ru/) для построения блок-схем	Алгоритм задачи построен по правилам формирования блок-схем и/или алгоритмического языка в соответствии с МУ к ПР 1.1-3.4	ОК 01, 02, 04, 05,09,10, ПК 1.1-1.2, ПК 2.5
У-5 Работать в среде программирования.	Использование интегрированные среды программирования: PyScripter, Wing IDE, PyCharm, интегрированную среду разработки для совместной работы в браузере repl.it(https://replit.com) для создания и тестирования программ	Программный код соответствует правилам хорошего стиля программирования и требованиям, определенным в МУ к ПР 2.1-4.14	ОК 01, 02, 04, 05,09,10, ПК 1.1-1.5, ПК 2.4, 2.5

У-6 Реализовывать построенные алгоритмы в виде программ на конкретном языке программирования.	Создание программы на языке высокого уровня python в интегрированных средах программирования: PyScripter, Wing IDE, PyCharm, интегрированную среду разработки для совместной работы в браузере repl.it(https://replit.com)	Программный код соответствует правилам синтаксиса и семантики языка python, и требованиям, определенным в МУ к ПР 2.1-4.14	ОК 01, 02, 04, 05,09,10, ПК 1.1-1.5, ПК 2.4, 2.5
У-7 Выполнять проверку, отладку кода программы.	Выполнение проверки программного кода, и отладка программного кода средствами интегрированной среды программирования, подготовка тестовых наборов	Отсутствие сообщений об ошибках, получение прогнозируемых результатов исполнения программного кода, согласно тестовых наборов и требованиям, определенным в МУ к ПР 2.1-4.14	ОК 01, 02, 04, 05,09,10, ПК 1.1-1.5, ПК 2.4, 2.5
З-1 Сквозные цифровые технологии и субтехнологии. Этические проблемы цифровизации. Цифровые риски и безопасность.	Перечисление сквозных цифровых технологий и субтехнологий, этических проблем и цифровых рисков.	Приведен полный список сквозных технологий. Перечислены субтехнологии для каждой технологии в соответствие с требованиями, определенным в МУ к ТЗ 1	ОК 01, 02, 04, 05,09,10,
З-2 Понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции.	Определение общих принципов построения алгоритмов, перечисление и краткая характеристика основных алгоритмических конструкций языка программирования	Алгоритм задачи построен по правилам формирования блок-схем и/или алгоритмического языка в соответствии с МУ к ПР 1.1-4.14	ОК 01, 02, 04, 05,09,10, ПК 1.1-1.5, ПК 2.5
З-3 Эволюцию языков программирования, их классификацию, понятие системы программирования.	Перечисление этапов развития языков программирования, классификации языков программирования. Определение понятия системы программирование.	Последовательность этапов правильная, языки программирования соответствуют своему классу в соответствие с требованиями, определенным в МУ к ТЗ 2	ОК 01, 02, 04, 05,09,10, ПК 1.1-1.2

3-4 Основные элементы языка, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти.	Перечисление и характеристика основных элементов процедурного языка программирования, структуры программы, операторов и операций, управляющих структур, структур данных, файлов, классов памяти.	Приведен полный список основных типов данных процедурного языка программирования, программный код содержит операторы и операции, управляющие структуры, структуры данных в соответствии с требованиями, определенным в МУ к ПР 1.1-4.14	ОК 01, 02, 04, 05,09,10, ПК 1.1-1.5, ПК 2.4, 2.5
3-5 Подпрограммы, составление библиотек подпрограмм	Определение правил создания подпрограммы и построения библиотеки программ	Приведена результативная последовательность шагов создания и использования подпрограммы; определена структура библиотеки подпрограмм в соответствии с учебно - методической литературой	ОК 01, 02, 04, 05,09,10, ПК 1.1-1.5, ПК 2.4, 2.5
3-6 Понятия искусственный интеллект, машинное обучение, технологии создания чат-ботов.	Определение понятий искусственный интеллект, машинное обучение, перечисление алгоритмов сравнения текстов, определение векторизации, правил обучения машинной модели, способов измерения качества классификаторов	Последовательность и правила создания чат-бота описаны правильно в соответствии с МУ к ТЗ 3	ОК 01, 02, 04, 05,09,10
3-7 Объектно-ориентированную модель программирования, основные принципы объектно-ориентированного программирования на примере алгоритмического языка: понятие классов и объектов, их свойств и методов, инкапсуляция и полиморфизма, наследования и переопределения алгоритмические конструкции.	перечисление основных принципов объектно-ориентированного программирования, определение структуры объявления классов, объектов, их свойств и методов.	Перечислены точно все основные принципы объектно-ориентированного программирования и дана их краткая характеристика полно и точно определена структура класса, содержащего не менее одного свойства и одного метода; создан объект класса для демонстрации обращения к методу и свойству класса в соответствии с учебно-методической литературой, в соответствии с требованиями, определенным в МУ к ПР 4.1-4.14	ОК 01, 02, 04, 05,09,10, ПК 1.1-1.5, ПК 2.4, 2.5

3. РАСПРЕДЕЛЕНИЕ ОЦЕНИВАНИЯ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ ПО ВИДАМ КОНТРОЛЯ

Наименование элемента умений или знаний	Виды аттестации	
	Текущий контроль	Промежуточная аттестация
У-1 Разрабатывать алгоритмы для конкретных задач.	+	+
У-2 Использовать языки программирования	+	+
У-3 Использовать платформы, онлайн сервисы и конструкторы сквозных цифровых технологий	+	
У-4 Использовать программы для графического отображения алгоритмов.	+	+
У-5 Работать в среде программирования.	+	+
У-6 Реализовывать построенные алгоритмы в виде программ на конкретном языке программирования.	+	+
У-7 Выполнять проверку, отладку кода программы.	+	+
3-1 Сквозные цифровые технологии и субтехнологии. Этические проблемы цифровизации. Цифровые риски и безопасность.	+	+
3-2 Понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции.	+	+
3-3 Эволюцию языков программирования, их классификацию, понятие системы программирования.	+	
3-4 Основные элементы языка, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти.	+	+
3-5 Подпрограммы, составление библиотек подпрограмм	+	
3-6 Понятия искусственный интеллект, машинное обучение, технологии создания чат-ботов.	+	
3-7 Объектно-ориентированную модель программирования, основные принципы объектно-ориентированного программирования на примере алгоритмического языка: понятие классов и объектов, их свойств и методов, инкапсуляция и полиморфизма, наследования и переопределения алгоритмические конструкции.	+	+

4.РАСПРЕДЕЛЕНИЕ ТИПОВ КОНТРОЛЬНЫХ ЗАДАНИЙ ПО ЭЛЕМЕНТАМ ЗНАНИЙ И УМЕНИЙ

Содержание учебного материала по программе УД	Тип контрольного задания													
	У1	У2	У3	У4	У5	У6	У7	31	32	33	34	35	36	37
Введение. Мировые цифровые тренды								Т, С						
Раздел 1. Введение в программирование														
Тема 1.1. Языки программирования	Пр	Пр		Пр					Пр	Т				
Тема 1.2. Типы данных	Пр	Пр		Пр					Пр		Т			
Раздел 2. Операторы языка программирования														
Тема 2.1 Операторы языка программирования python	Пр	Пр	Пр	Пр	Пр	Пр	Пр		Пр			Т		Пр
Раздел 3. Структурное и модульное программирование														
Тема 3.1. Процедуры и функции	Пр	Пр	Пр	Пр	Пр	Пр	Пр		Пр					Пр
Тема 3.2. Структуризация в программировании	Пр	Пр			Пр	Пр	Пр		Пр					Пр
Тема 3.3. Модульное программирование	Пр	Пр			Пр	Пр	Пр		Пр			Пр		Пр
Раздел 4. Объектно-ориентированное программирование														
Тема 4.1 Основные принципы объектно-ориентированного программирования (ООП)	Пр	Пр			Пр	Пр	Пр		Пр					Т, Пр
Тема 4.2. Интегрированная среда разработчика	Пр	Пр			Пр	Пр	Пр		Пр					Пр
Тема 4.3. Визуальное событийно-управляемое программирование	Пр	Пр			Пр	Пр	Пр		Пр					Пр
Тема 4.4 Разработка оконного приложения	Пр	Пр			Пр	Пр	Пр		Пр					Пр
Тема 4.5 Этапы разработки приложений	Пр	Пр			Пр	Пр	Пр		Пр					Пр
Тема4.6 Иерархия классов														

Пр- практическая работа, Т – тест, С- семинар

5 РАСПРЕДЕЛЕНИЕ ТИПОВ И КОЛИЧЕСТВА КОНТРОЛЬНЫХ ЗАДАНИЙ ПО ЭЛЕМЕНТАМ ЗНАНИЙ И УМЕНИЙ, КОНТРОЛИРУЕМЫХ НА ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

Содержание учебного материала по программе УД	Тип контрольного задания													
	У1	У2	У3	У4	У5	У6	У7	З1	З2	З3	З4	З5	З6	З7
Введение. Мировые цифровые тренды								Т						
Раздел 1. Введение в программирование														
Тема 1.1. Языки программирования	Т													
Тема 1.2. Типы данных	Т													
Раздел 2. Операторы языка программирования														
Тема 2.1 Операторы языка программирования python		Т		Т	Пр	Пр	Пр		Т, Пр		Т, Пр			
Раздел 3. Структурное и модульное программирование														
Тема 3.1. Процедуры и функции		Т		Т	Пр	Пр	Пр							
Тема 3.2. Структуризация в программировании														
Тема 3.3. Модульное программирование														
Раздел 4. Объектно-ориентированное программирование														
Тема 4.1 Основные принципы объектно-ориентированного программирования (ООП)		Т												Т
Тема 4.2. Интегрированная среда разработчика														
Тема 4.3. Визуальное событийно-управляемое программирование														
Тема 4.4 Разработка оконного приложения														
Тема 4.5 Этапы разработки приложений														
Тема 4.6 Иерархия классов														

В соответствии со ст. 43 Конституции Российской Федерации, 273-ФЗ «Об образовании в Российской Федерации» от 29.12.2012, приказом Минобрнауки России от 09.11.2015 N 1309 «Об утверждении Порядка обеспечения условий доступности для инвалидов объектов и предоставляемых услуг в сфере образования, а также оказания им при этом необходимой помощи», ГОСТ Р 57723-2017 «Информационно-коммуникационные технологии в образовании. Системы электронно-библиотечные. Общие положения», ГОСТ Р 52872-2019 «Интернет-ресурсы и другая информация, представленная в электронно-цифровой форме. Приложения для стационарных и мобильных устройств, иные пользовательские интерфейсы. Требования доступности для людей с инвалидностью и других лиц с ограничениями жизнедеятельности», все предлагаемые электронные ресурсы максимально комфортны для чтения слабовидящими людьми. Масштабирование текста достигает 300 процентов. При изменении масштаба сохраняется возможность видеть всю страницу текста, не обрезая его.

6. СТРУКТУРА КОНТРОЛЬНЫХ ЗАДАНИЙ

6.1. Презентации по предложенным темам:

- 1) Понятие цифровых технологий и цифровой экономики.
- 2) Информационный продукт как результат цифровой экономики.
- 3) Структура и тенденции развития рынка цифровых технологий.
- 4) Этические проблемы цифровизации.
- 5) Цифровые риски и безопасность.
- 6) Вызовы и угрозы цифровой экономики.
- 7) Программа "Цифровая экономика Российской Федерации" как основополагающий документ.
- 8) Предпосылки развития цифровой экономики Российской Федерации. Цели и задачи программы.
- 9) Базовые направления развития и сквозные цифровые технологии.
- 10) Системы распределенного реестра. Блок-чейн.
- 11) Интернет вещей. Big-data
- 12) Нейротехнологии и искусственный интеллект.
- 13) Квантовые технологии.
- 14) Большие данные для решения прикладных задач.
- 15) Компоненты робототехники и сенсорики.
- 16) Технологии беспроводной связи.
- 17) Технологии виртуальной и дополненной реальностей.
- 18) Сферы применения сквозных технологий.

Шкала оценки достижений

Процент результативности	Кол-во баллов	Оценка уровня подготовки	
		Отметка	Вербальный аналог
90 ÷ 100	18-20	5	отлично
75 ÷ 89	15-17	4	хорошо
60 ÷ 74	12-14	3	удовлетворительно
менее 60	Менее 12	2	неудовлетворительно

6.1 Тест по теме «Системы программирования»

1. Отметьте все языки программирования высокого уровня.
 - Паскаль
 - Си
 - язык ассемблера
 - машинные коды
 - python
2. Отметьте языки программирования, которые используются для создания Web-сайтов в Интернете.
 - Javascript
 - Паскаль
 - PHP
 - Си
 - Perl
3. Как называется программа, которая переводит в машинные коды тексты программ, написанных на языке высокого уровня?
Ответ: введите строку
4. Как называется программа, которая переводит в машинные

коды символьную запись машинных команд?

Ответ: введите строку

5. В чем преимущества трансляторов-интерпретаторов в сравнении с компиляторами?

- программы работают быстрее
- программы легче отлаживать
- для выполнения программы не надо устанавливать транслятор
- программы работают надежнее
- программы переносимы на любую платформу, для которой есть интерпретатор

6. Назовите тип транслятора, который переводит в машинный код сразу всю программу и строит исполняемый файл.

Ответ: введите строку

7. Как называется промежуточный исполняемый код, который выполняется виртуальной машиной?

Ответ: введите строку

8. Как называется программа, которая собирает разные части (модули) создаваемой программы и функции из стандартных библиотек в единый исполняемый файл?

Ответ: введите строку

9. Как называется программа для поиска ошибок в других программах?

Ответ: введите строку

10. Как называется программа, позволяющая оценить время работы каждой процедуры и функции во время выполнения программы?

Ответ: введите строку

11. Каким термином называется набор стандартных структур данных и функций операционной системы или сервиса, которые программисты могут использовать в своих программах?

- интерфейс программирования приложений
- программный интерфейс
- список процедур и функций
- описание структур данных и функций
- порядок вызова стандартных функций

12. Отметьте все программы, которые обычно входят в состав интегрированной среды разработки программ (IDE).

- текстовый редактор
- компоновщик
- отладчик
- профилировщик
- табличный процессор

13. Как называются среды программирования, нацеленные на высокую скорость и низкую стоимость разработки программ?

- среды быстрой разработки приложений
- среды разработки надёжных приложений
- среды разработки дешёвых приложений
- среды проектирования приложений

14. Отметьте языки программирования, программа на которых может быть переведена в псевдокод, выполняемый в виртуальной машине.

- Visual Basic
- C#
- Java
- python
- Паскаль

Шкала оценки достижений

Процент результативности	Кол-во баллов	Оценка уровня подготовки	
		Отметка	Вербальный аналог
90 ÷ 100	9-10	5	отлично
75 ÷ 89	7-8	4	хорошо
60 ÷ 74	5-6	3	удовлетворительно
менее 60	Менее 5	2	неудовлетворительно

6.3. Тест «Искусственный интеллект и машинное обучение»

1. Что послужило моделью для искусственной нейронной сети?

- процессы головного мозга
- паутиная сеть
- компьютерная сеть
- телефонная сеть

2. Способна ли искусственная нейронная сеть обучаться?

- Да
- Нет

3. Лучший игрок в шахматы:

- Антон Сихарулидзе
- искусственный интеллект
- Гарри Каспаров

4. Система видеонаблюдения с применением оптико-электронных приспособлений, которые предназначены работы в автоматическом режиме с анализом получаемых изображений, искусственный интеллект?

- Да
- Нет

5. Компании Cognitive Technologies и Камский автозавод «КамАЗ» объявили о начале проекта по созданию беспилотного грузовика. Он с ИИ?

- Да
- Нет

6. Чтобы найти что-то в интернете, мы формируем запрос. Поисковая система находит нужную информацию по ключевым словам. Информационный поиск относится к обработке естественного языка. Поисковая система - ИИ?

- Да
- Нет

7. Что из перечисленного не является ИИ?

- Компьютерное зрение
- Экспертная система
- Обработка текста на естественном языке

8. Искусственный интеллект в современном образовательном процессе:

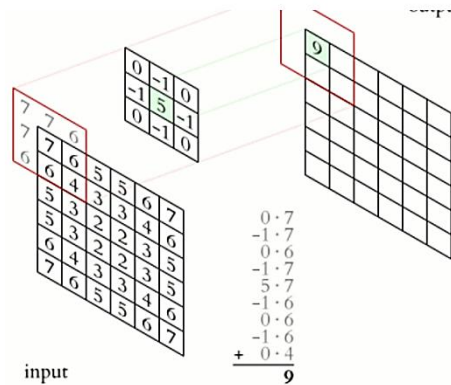
- Не внедряется
- Планируется к внедрению
- Используется в некоторых предметных областях
- Может быть внедрён в полной мере

9. Активное изучение ИИ в разные периоды жизни общества относятся к:

- 1-й промышленной революции
- 2-й промышленной революции
- 3-й промышленной революции
- 4-й промышленной революции

10. Какой из приведённых терминов характеризует ИИ, способный к эмоциям и являющийся подобным человеческому интеллекту?
 - Слабый ИИ
 - Развитый ИИ
 - Сильный ИИ
 - Умный ИИ
11. Искусственные нейронные сети (ИНС) — модели машинного обучения, использующие комбинации распределенных простых операций, зависящих от обучаемых параметров, для обработки входных данных. Какого вида ИНС не существует?
 - Противоборствующие
 - Рекуррентные
 - Наивные
 - Импульсные
12. У машинного обучения есть ряд задач. Как называется та, что направлена на предсказание значения той или иной непрерывной числовой величины для входных данных?
 - Кластеризация
 - Регрессия
 - Переобучение
 - Классификация
13. Особых успехов нейросети достигли в работе с изображениями. Но что из этого нейросети не могут сделать?
 - Стилизовать вашу фотографию под работу импрессиониста
 - Омолаживать и состаривать лица на фотографиях
 - Догадаться, что вы нарисовали
 - Пластическую коррекцию лица
14. Кто создал первую модель искусственных нейронных сетей?
 - Ян Лекун
 - Фрэнк Розенблатт
 - Дэвид И. Румельхарт, Дж. Е. Хинтон и Рональд Дж. Вильямс
 - Мак-Каллок и Питтс
15. Какой из видов машинного обучения основывается на взаимодействии обучаемой системы со средой?
 - Обучение без учителя
 - Обучение с подкреплением
 - Обучение с учителем
 - Глубинное обучение
16. Когда говорят о нейронных сетях и машинном обучении, часто упоминают закон Мура. В чем его суть?
 - Не следует множить сущее без необходимости
 - 20% усилий дают 80% результата, а остальные 80% усилий — лишь 20% результата
 - Каждое следующее поколение компьютеров работает в 2,5 раза быстрее
 - Если все слова языка или длинного текста упорядочить по убыванию частоты их использования, то частота n-го слова в таком списке окажется приблизительно обратно пропорциональной его порядковому номеру n
17. В какие игры нейросеть еще не научилась обыгрывать человека?
 - Бридж
 - Шахматы
 - «Марио»
 - Го

18. Допустим, нам нужно рассчитать необходимые параметры для создания обшивки самолета. Какая из областей машинного обучения нам в этом пригодится?
- Обучение ранжированию
 - Компьютерное зрение
 - Латентная модель
 - Предсказательное моделирование
19. Какой тип искусственной нейронной сети представлен на картинке?



- Сверточная нейронная сеть
- Рекуррентная нейронная сеть
- Нейронная сеть Джордана
- Простая нейронная сеть

Шкала оценки достижений

Процент результативности	Кол-во баллов	Оценка уровня подготовки	
		Отметка	Вербальный аналог
90 ÷ 100	9-10	5	отлично
75 ÷ 89	7-8	4	хорошо
60 ÷ 74	5-6	3	удовлетворительно
менее 60	Менее 5	2	неудовлетворительно

6.4 Практическая работа по теме 2.1.5 «Списки и кортежи»

Цель работы: применение теоретических знаний и демонстрация умений в решении задачи с использованием типов данных: списки и кортежи

Методические указания

1. Прочитайте Резюме по теме «Списки и кортежи».
2. Внимательно ознакомьтесь с 3 заданиями лабораторной работы.
3. Изучите предложенные рекомендации для разработки программного кода задания.
4. Напишите программный код с указанием комментариев построчно в PyScripter.
5. Исполните программный код на разных наборах данных (ожидаемый результат, ошибочный результат, непредсказуемый результат).
6. Сохраните исполняемый программный код на своем компьютере.
7. Оформите результат кодирования и исполнения копиями экрана.
8. После выполнения всех 5 заданий, оформите отчет с титульным листом, номером задания, содержанием задачи, подготовленными копиями экрана.
9. Сохраните отчет, отметив около кнопки ОК, сервис-> сжать рисунки-> электронная почта 96 пикселей на дюйм.

При нарушении этого требования, оценка снижается на 1 балл.

10. Если будут учтены «уязвимости кода» и предложены решения, то дополнительно назначается по 2 балла за каждое устранение уязвимости.

11. Отправить отчет на проверку через LMS Moodle («рука» Лабораторная работа по теме 1.2.5.)

Максимальная оценка за работу 15 баллов, дополнительно можно набрать 6 баллов

Задание 1

1. Напишите программу, в которой на основе текста, введенного пользователем, создается кортеж. Затем на основе этого кортежа создается новый кортеж. В новый кортеж включаются равноотстоящие элементы, начиная с первого (с нулевым индексом). Например, в новый кортеж включаются элементы, отстоящие друг от друга на 3 позиции (элементы с индексами 0, 3, 6, 9 и так далее). Расстояние между элементами (приращение по индексу) вводится пользователем.

Рекомендации:

Сообщить пользователю, что необходимо ввести текст. С помощью функции `input()` вводите в программу текст, присваивая значение строковой переменной

С помощью функции `tuple()` создаете кортеж, например `C`.

Запрашиваете шаг дискретности по индексу. (`k=input('Введите.....')`).
Помните, что введенное значение – это строка.

Преобразуем строку в целое с помощью функции `n=int(строковая переменная)`.

Организуем «срез» с кортежа `C` в новый кортеж `B`. (срез `C[:n]` означает, что выбираются из всего кортежа элементы через шаг `n`).

Далее формируем символьную строку `k=str(n)`

Печатаем результат (`print ('Элементы кортежа с шагом '+k+':', указываем переменную – полученный кортеж, end='')`)

*Так как необходимо ввести целое число, то стоит избежать ввода действительного числа. (стр 79 Учебника Васильева по Питону)

Задание 2

2. Напишите программу, в которой пользователь вводит целое число, а программа формирует кортеж, который состоит из цифр, входящих в это число. Предложите способы создания кортежа, при котором цифры, формирующие число, включаются в кортеж в прямом и обратном порядке.

Рекомендации:

Учитываем, что с клавиатуры вводится текстовый формат- не нужно преобразовывать числовое значение в строковое `n=input('Введите.....')`

*Так как необходимо ввести целое число, то стоит избежать ввода действительного числа, или символьного (стр 79 Учебника Васильева по Питону)

Формируем кортеж, используя функции `tuple()` : возможно, `A=tuple(n)` (проверьте)

Организуем «срез» с кортежа в новый кортеж: `C= A[:]`

Печатаем результат (`print ('Прямой кортеж ',C,end='« »')`)

Организуем «срез в обратном порядке» с кортежа в новый кортеж `D`, используя срез в обратном порядке с шагом `-1` `A[-1::-1]`

Печатаем результат (`print ('Обратный кортеж ',D,end='')`)

Задание 3

3. Напишите программу, в которой создается числовой список. Список заполняется случайными числами. Затем элементы с четными индексами сортируются в порядке возрастания, а элементы с нечетными индексами сортируются в порядке убывания.

Рекомендации:

1. Организуем случайный ввод числа самой программой. Подключаем библиотеку `from random import *`

`Seed(2022)` - Инициализация генератор

2. Создаем список случайных чисел

`A=[randint(10,20) for k in range(15)]`

*Пример.(стр 165 Учебника Васильева по Питону)

3. Создаем пустые списки `B=[],C=[]`

4. Печатаем результат (`print ('Случайный список',A,end='')`)

Находим остаток от деления `k=n%2`

`for n in range (0,len(A)-1):`

`if k==0 :`

`C=A[n] #ЧЕТНЫЕ`

`else:`

`B=A[n] # НЕЧЕТНЫЕ`

*Объясните в комментариях, что происходит

`B.sort(revers=True)`

`D=sorted(C)`

`p=int(long(A)/2)`

`for n in range (0,p-1):`

`A=D[n]+B[n]`

Печатаем результат (`print ('Отсортированный список',A,end='« »)`)

Шкала оценки достижений

Процент результативности	Кол-во баллов	Оценка уровня подготовки	
		Отметка	Вербальный аналог
90 ÷ 100	9-10	5	отлично
75 ÷ 89	7-8	4	хорошо
60 ÷ 74	5-6	3	удовлетворительно
менее 60	Менее5	2	неудовлетворительно

6.5 Практическая работа Тема 3.3 Модульное программирование

На основании презентаций, лекций, статей выполните задания в виде опорного конспекта, используя цифровые инструменты (<https://sboard.online/> -Электронная доска, <https://prezi.com/> Prezi Мультимедия в презентациях):

1. Дайте определение модуля, которое для вас наиболее понятно.

2. Опишите структуру и основные свойства модуля

3. Составьте таблицу с указанием сильных и слабых сторон модульного программирования.

4. Проанализировать требования к организации структуры модуля на разных языках программирования (лекции и статьи)

5. Сделайте свой вывод.

Шкала оценки достижений

Процент	Кол-во баллов	Оценка уровня подготовки
---------	---------------	--------------------------

результативности		Отметка	Вербальный аналог
90 ÷ 100	7-8	5	отлично
75 ÷ 89	5-6	4	хорошо
60 ÷ 74	3-4	3	удовлетворительно
менее 60	Менее3	2	неудовлетворительно

7. ПРОМЕЖУТОЧНЫЙ КОНТРОЛЬ ЗНАНИЙ

Промежуточный контроль знаний проводится в форме экзамена

Оценка освоения дисциплины предусматривает использование системы оценивания путем подсчитывания всех баллов по дисциплине, наличие положительных оценок, наличие конспекта по теоретическим занятиям, выполненных практических работ.

7.1 Вопросы к экзамену

1. Жизненный цикл программы. Программный продукт и его характеристики.
2. Понятие алгоритма. Свойства алгоритма. Способы записи алгоритма.
3. Принципы структурного программирования. Базовые алгоритмические структуры: следование, ветвление, цикл. Метод пошаговой детализации разработки алгоритмов.
4. Языки программирования. Классификация языков программирования. Понятие уровня языка программирования.
5. Системы программирования.
6. Методы трансляции программ: компиляция, интерпретация.
7. Язык программирования python: особенности и области применения. Преимущества и недостатки языка python
8. Требования к среде разработки на python. Интегрированные среды разработки и редакторы для языка python.
9. Структура python-программ. Идентификаторы и ключевые слова. Правила именования переменных.
10. Изменяемые и неизменяемые типы данных python. Встроенные типы данных в python: скалярные типы, коллекции.
11. Числовые типы данных в python: целочисленные, числа с плавающей точкой. Операции с числами.
12. Функции модуля math стандартной библиотеки python. Генерация случайных чисел модуль random.
13. Последовательности в python, основные признаки и типы последовательностей.
14. Строковый (str) тип данных в python. Операции со строками: индексация, срез, конкатенация, повторение. Строковые методы
15. Кортежи (tuple) в python. Основные операции со кортежами
16. Списки (list) в python. Основные операции, методы и инструкции
17. Множества (set) в python. Отличие множества от других типов коллекций. Основные методы и операторы множеств. Фиксированное множество (frozenset).
18. Словари (dict) в python. Операторы и основные методы словарей
19. Файловый объект в python. Функция open(), режимы доступа к файлам. Назначение методов файлового объекта: read(), readline(), write() и writeline(), close().
20. Управляющие структуры python: ветвление (инструкция if), циклическая обработка (инструкции while и for ... in). Особенности применения инструкций continue, break, return.

21. Понятие функции, описание функции в python. Типы функций в python: глобальные, локальные, лямбда функции, методы.
22. Модули и пакеты в python. Специальные и дополнительные атрибуты модуля. Инструкция import. Назначение функций dir() и help().
23. Базовые понятия ООП: объект, его свойства и методы, класс, интерфейс.
24. Основные принципы ООП: инкапсуляция, наследование, полиморфизм.
25. ООП на python: классы, экземпляры классов, атрибуты, методы. Назначение функции hasattr().
26. ООП на python: создание класса, инициализация класса, создание экземпляра класса (__new__), конструктор (__init__), деструктор (__del__) в python.
27. Этапы разработки объектно-ориентированного приложения.
28. Создание интерфейса пользователя.
29. Программирование приложения. Тестирование и отладка приложения.
30. Создание документации. Понятие дружественного интерфейса. Организация интерфейса программы.

7.2 Экзаменационные билеты

На выполнение экзаменационной работы дается 2 часа (120 минут). Работа состоит из 2 частей. Часть 1 .Теоретическая часть – тестирование . Часть 2 Практическое задание.

1. Вопросы экзаменационного теста с ответами

Вопрос 1

Установите соответствие между типа данных, которые потенциально можно преобразовать в целое число напрямую через функцию int() и возвращаемым ею результатом:			MAT
Балл по умолчанию:			2
Перемешать:			Да
Показать количество правильных ответов после окончания:			Да
Штраф за каждую неправильную попытку:			0
ID-номер:			
#	Вопрос	Ответ	
1.	целые числа: int(555) =	555	
2.	числа с плавающей точкой: int(-32.45) =	-32	
3.	булевы значения: int(False) =	0	
4.	десятичные дроби: int(Decimal('17.7'))	17	
5.	дробные числа: int(Fraction(10, 7))=	1	
6.	строки: int('31') =	31	
7.	строки: int('-32,95') =	-32	
8.		-33	

Установите соответствие между типа данных, которые потенциально можно преобразовать в целое число напрямую через функцию <code>int()</code> и возвращаемым ею результатом:		<i>MAT</i>
Балл по умолчанию:		2
Перемешать:		Да
Показать количество правильных ответов после окончания:		Да
Штраф за каждую неправильную попытку:		0
ID-номер:		
#	Вопрос	Ответ
	Общий отзыв к вопросу:	
	Для любого правильного ответа:	Ваш ответ верный.
	Для любого неправильного ответа:	Ваш ответ неправильный.
	Для любого частично правильного ответа:	Ваш ответ частично правильный.
	Подсказка 1:	
	Показать количество правильных ответов (Подсказка 1):	Нет
	Удалить некорректные ответы (Подсказка 1):	Нет
	Теги:	
<i>Вы должны указать по меньшей мере два вопроса и три ответа. Вы можете включить дополнительные неправильные ответы, создав ответ на пустой вопрос. Записи, где и вопрос и ответ пустые, будут игнорироваться</i>		

Вопрос 2

Укажите основные свойства кортежей			МА
Балл по умолчанию:			1
Случайный порядок ответов:			Да
Нумеровать варианты ответов?:			1
Показать количество правильных ответов после окончания:			Да
Штраф за каждую неправильную попытку:			10
ID-номер:			
#	Ответы	Отзыв	Оценка
A.	Они неизменяемы (нельзя добавить, удалить новые элементы)	правильно	25
B.	Они изменяемы (можно добавить, удалить новые элементы)	неправильно	0
C.	Это упорядоченные коллекции (всегда известно, на какой позиции какой элемент находится)	правильно	25
D.	Это неупорядоченные коллекции (неизвестно, на какой позиции какой элемент находится)	неправильно	0
E.	Длина однажды созданного кортежа не меняется никогда	правильно	25
F.	Применяются для формирования контейнера разнородных объектов	правильно	25
G.	Длину созданного кортежа можно изменить	неправильно	0
Общий отзыв к вопросу:			
Для любого правильного ответа:		Ваш ответ верный.	
Для любого неправильного ответа:		Ваш ответ неправильный.	
Для любого частично правильного ответа:		Ваш ответ частично правильный.	
Подсказка 1:			
Показать количество правильных ответов (Подсказка 1):		Нет	
Удалить некорректные ответы (Подсказка 1):		Нет	
Теги:			
<i>Позволяет выбирать один или несколько правильных ответов из заданного списка. (MC/MA)</i>			

Вопрос 3

Перечислите способы создания кортежей.			МА
Балл по умолчанию:			1
Случайный порядок ответов:			Да
Нумеровать варианты ответов?:			1
Показать количество правильных ответов после окончания:			Да
Штраф за каждую неправильную попытку:			10
ID-номер:			
#	Ответы	Отзыв	Оценка
A.	Через конструктор tuple	правильно	33.3
B.	При помощи круглых скобок	правильно	33.3
C.	С помощью запятых	правильно	33.3
D.	При помощи квадратных скобок	неправильно	0
Общий отзыв к вопросу:			
Для любого правильного ответа:		Ваш ответ верный.	
Для любого неправильного ответа:		Ваш ответ неправильный.	
Для любого частично правильного ответа:		Ваш ответ частично правильный.	
Подсказка 1:			
Показать количество правильных ответов (Подсказка 1):		Нет	
Удалить некорректные ответы (Подсказка 1):		Нет	
Теги:			
<i>Позволяет выбирать один или несколько правильных ответов из заданного списка. (МС/МА)</i>			

Вопрос 4

Выберите верные утверждения.			МА
Балл по умолчанию:			1
Случайный порядок ответов:			Да
Нумеровать варианты ответов?			1
Показать количество правильных ответов после окончания:			Да
Штраф за каждую неправильную попытку:			10
ID-номер:			
#	Ответы	Отзыв	Оценка
A.	Мощность множества показывает количество элементов в нем.	правильно	33.3
B.	Функция len() применяется к множествам для вычисления T мощности множества	правильно	33.3
C.	Мощность множества (Интерактивный режим >>> st = {3, 4, 8, 89} >>> len(st) равна 4	правильно	33.3
D.	Мощность множества (Интерактивный режим >>> lst = [1, 5, 4, 5, 5] >>> len(lst) равна 5	неправильно	0
Общий отзыв к вопросу:			
Для любого правильного ответа:		Ваш ответ верный.	
Для любого неправильного ответа:		Ваш ответ неправильный.	
Для любого частично правильного ответа:		Ваш ответ частично правильный.	
Подсказка 1:			
Показать количество правильных ответов (Подсказка 1):		Нет	
Удалить некорректные ответы (Подсказка 1):		Нет	
Теги:			
<i>Позволяет выбирать один или несколько правильных ответов из заданного списка. (MC/MA)</i>			

Вопрос 5

Удалить элемент множества не вызывая ошибку, если его там нет, можно способами:			МА
Балл по умолчанию:			1
Случайный порядок ответов			Да
Нумеровать варианты ответов?			1
Показать количество правильных ответов после окончания:			Да
Штраф за каждую неправильную попытку:			10
ID-номер:			
#	Ответы	Отзыв	Оценка
A.	pop() – удаляет случайный элемент и возвращает его, но вызовет ошибку, если множество пустое	неправильно	0
B.	remove() – удаляет объект с конкретным значением, но вызовет ошибку, если его нет	неправильно	0
C.	discard() – также удаляет определенный элемент, но не приводит к ошибке, если он отсутствует в данном множестве	правильно	100
Общий отзыв к вопросу:			
Для любого правильного ответа:		Ваш ответ верный.	
Для любого неправильного ответа:		Ваш ответ неправильный.	
Для любого частично правильного ответа:		Ваш ответ частично правильный.	
Подсказка 1:			
Показать количество правильных ответов (Подсказка 1):		Нет	
Удалить некорректные ответы (Подсказка 1):		Нет	
Теги:			
Позволяет выбирать один или несколько правильных ответов из заданного списка. (MC/MA)			

Вопрос 6

Что представляет собой супермножество?			МА
Балл по умолчанию:			1
Случайный порядок ответов:			Да
Нумеровать варианты ответов?			1
Показать количество правильных ответов после окончания:			Да
Штраф за каждую неправильную попытку:			10
ID-номер:			
#	Ответы	Отзыв	Оценка
A.	Супермножество включает в себя все элементы другого множества и может быть равно с ним по мощности	верно	25
B.	Для проверки на супермножество можно использовать метод <code>issuperset()</code>	верно	25
C.	Для проверки на супермножество можно использовать знак <code>>=</code>	верно	25
D.	Чистое супермножество, строго больше подмножества и содержит все его компоненты	верно	25
Общий отзыв к вопросу:			
Для любого правильного ответа:		Ваш ответ верный.	
Для любого неправильного ответа:		Ваш ответ неправильный.	
Для любого частично правильного ответа:		Ваш ответ частично правильный.	
Подсказка 1:			
Показать количество правильных ответов (Подсказка 1):		Нет	
Удалить некорректные ответы (Подсказка 1):		Нет	
Теги:			
<i>Позволяет выбирать один или несколько правильных ответов из заданного списка. (MC/MA)</i>			

Вопрос 7

Установите верное соответствие			МАТ
Балл по умолчанию:			1
Перемешать:			Да
Показать количество правильных ответов после окончания:			Да
Штраф за каждую неправильную попытку:			0
ID-номер:			
#	Вопрос	Ответ	
1.	Метод удаляет случайный элемент и возвращает его, но вызовет ошибку, если множество пустое	pop()	
2.	Метод удаляет объект с конкретным значением, но вызовет ошибку, если его нет	remove()	
3.	Метод удаляет определенный элемент, но не приводит к ошибке, если он отсутствует в данном множестве	discard()	
4.		union	
Общий отзыв к вопросу:			
Для любого правильного ответа:		Ваш ответ верный.	
Для любого неправильного ответа:		Ваш ответ неправильный.	
Для любого частично правильного ответа:		Ваш ответ частично правильный.	
Подсказка 1:			
Показать количество правильных ответов (Подсказка 1):		Нет	
Удалить некорректные ответы (Подсказка 1):		Нет	
Теги:			
<p><i>Вы должны указать по меньшей мере два вопроса и три ответа. Вы можете включить дополнительные неправильные ответы, создав ответ на пустой вопрос. Записи, где и вопрос и ответ пустые, будут игнорироваться</i></p>			

Вопрос 8

Опишите способы импортирования:			МА
Балл по умолчанию:			1
Случайный порядок ответов:			Да
Нумеровать варианты ответов?:			1
Показать количество правильных ответов после окончания:			Да
Штраф за каждую неправильную попытку:			10
ID-номер:			
#	Ответы	Отзыв	Оценка
A.	Через импорт библиотеки Интерактивный режим >>> import math >>> math.sqrt(25)	правильно	20
B.	Через импорт всех объектов модуля Интерактивный режим >>> from math import * >>> sqrt(25)	правильно	20
C.	Через импорт конкретной функции Интерактивный режим >>> from math import sqrt >>> sqrt(25)	правильно	20
D.	С помощью импорта функции с переименованием Интерактивный режим >>> from math import sqrt as square_root >>> square_root(25)	правильно	20
E.	С помощью импорта модуля с переименованием Интерактивный режим >>> import math as mathematics >>> mathematics.sqrt(25)	правильно	20
Общий отзыв к вопросу:			
Для любого правильного ответа:		Ваш ответ верный.	
Для любого неправильного ответа:		Ваш ответ неправильный.	
Для любого частично правильного ответа:		Ваш ответ частично правильный.	
Подсказка 1:			
Показать количество правильных ответов (Подсказка 1):		Нет	
Удалить некорректные ответы (Подсказка 1):		Нет	
Теги:			
Позволяет выбирать один или несколько правильных ответов из заданного списка. (MC/MA)			

Вопрос 9

Для чего необходимо ключевое слово self в классах?			МА
Балл по умолчанию:			1
Случайный порядок ответов:			Да
Нумеровать варианты ответов?			1
Показать количество правильных ответов после окончания:			Да
Штраф за каждую неправильную попытку:			10
ID-номер:			
#	Ответы	Отзыв	Оценка
A.	В качестве первого аргумента у методов экземпляра класса;	правильно	50
B.	5. Для доступа к свойству объекта внутри класса.	правильно	50
C.	Для создания статического метода	неправильно	0
Общий отзыв к вопросу:			
Для любого правильного ответа:		Ваш ответ верный.	
Для любого неправильного ответа:		Ваш ответ неправильный.	
Для любого частично правильного ответа:		Ваш ответ частично правильный.	
Подсказка 1:			
Показать количество правильных ответов (Подсказка 1):		Нет	
Удалить некорректные ответы (Подсказка 1):		Нет	
Теги:			
<i>Позволяет выбирать один или несколько правильных ответов из заданного списка. (MC/MA)</i>			

Вопрос 10

Какими способами можно получить доступ к значению ключа, не изменяя при этом словарь?			МА
Балл по умолчанию:			1
Случайный порядок ответов:			Да
Нумеровать варианты ответов?			1
Показать количество правильных ответов после окончания:			Да
Штраф за каждую неправильную попытку:			10
ID-номер:			
#	Ответы	Отзыв	Оценка
A.	<p>Стандартный способ доступа к значению словаря – через квадратные скобки</p> <p>Интерактивный режим</p> <pre>>>> my_dict = {'121': 'Значение 1', 'key2': 'Значение 2', 'key3': 'Значение 3'} >>> my_dict[121]</pre>	правильно	50
B.	<p>Метод возврата значения словаря, основанный на методе get().</p> <p>Интерактивный режим</p> <pre>>>> my_dict = {'k1': 120, 'k2': 'Nice key', 'k3': [1, 2, 99]} >>> my_dict.get('k2') Nice key >>> my_dict.get('k4') None >>> my_dict.get('k4', 'Нет такого ключа') 'Нет такого ключа' >>> my_dict {'k1': 120, 'k2': 'Nice key', 'k3': [1, 2, 99]}</pre>	правильно	50
Общий отзыв к вопросу:			
Для любого правильного ответа:		Ваш ответ верный.	
Для любого неправильного ответа:		Ваш ответ неправильный.	
Для любого частично правильного ответа:		Ваш ответ частично правильный.	
Подсказка 1:			
Показать количество правильных ответов (Подсказка 1):		Нет	
Удалить некорректные ответы (Подсказка 1):		Нет	
Теги:			
Позволяет выбирать один или несколько правильных ответов из заданного списка. (MC/MA)			

Вопрос 11

Перечислите структуры данных и общие требования к именованию.			МА
Балл по умолчанию:			1
Случайный порядок ответов			Да
Нумеровать варианты ответов?			1
Показать количество правильных ответов после окончания:			Да
Штраф за каждую неправильную попытку:			10
ID-номер:			
#	Ответы	Отзыв	Оценка
А.	Главное требование к имени ключа словаря в Python – он должен быть хешируем	правильно	12.5
В.	Хешируемость означает неизменяемость, немутуруемость.	правильно	12.5
С.	К немутабельным объектам относят числа	правильно	12.5
Д.	К немутабельным объектам относят: строки	правильно	12.5
Е.	К немутабельным объектам относят булевы значения	правильно	12.5
Ф.	К немутабельным объектам относят кортежи	правильно	12.5
Г.	К немутабельным объектам относят frozenset	правильно	12.5
Н.	В словаре не может быть двух ключей с одинаковыми наименованиями	правильно	12.5
Общий отзыв к вопросу:			
Для любого правильного ответа:		Ваш ответ верный.	
Для любого неправильного ответа:		Ваш ответ неправильный.	
Для любого частично правильного ответа:		Ваш ответ частично правильный.	
Подсказка 1:			
Показать количество правильных ответов (Подсказка 1):		Нет	
Удалить некорректные ответы (Подсказка 1):		Нет	
Теги:			
<i>Позволяет выбирать один или несколько правильных ответов из заданного списка. (MC/MA)</i>			

Вопрос 12

Установите соответствие между методами keys(), items(), values() и возвращенным результатом, спецификой результирующих объектов?			MAT
Балл по умолчанию:			1
Перемешать:			Да
Показать количество правильных ответов после окончания:			Да
Штраф за каждую неправильную попытку:			0
ID-номер:			
#	Вопрос	Ответ	
1.	Метод возвращает список ключей в исходном порядке, все элементы уникальны, ведут себя как множество (set): могут объединяться, пересекаться и т.п., являются итерабельными, их нельзя менять напрямую и вызывать по индексу	keys()	
2.	Метод - контейнер со значениями словаря, в котором элементы могут повторяться (т.е. не ведет себя как множество).	values()	
3.	Метод возвращает список элементов словаря в виде кортежей, где первым идет ключ, а вторым объектом – значение.	items()	
4.		views	
	Общий отзыв к вопросу:		
	Для любого правильного ответа:	Ваш ответ верный.	
	Для любого неправильного ответа:	Ваш ответ неправильный.	
	Для любого частично правильного ответа:	Ваш ответ частично правильный.	
	Подсказка 1:		
	Показать количество правильных ответов (Подсказка 1):	Нет	
	Удалить некорректные ответы (Подсказка 1):	Нет	
	Теги:		
<p><i>Вы должны указать по меньшей мере два вопроса и три ответа. Вы можете включить дополнительные неправильные ответы, создав ответ на пустой вопрос. Записи, где и вопрос и ответ пустые, будут игнорироваться</i></p>			

Вопрос 13

Перечислите характеристики типа данных «список», которые вы знаете.			МА
Балл по умолчанию:			1
Случайный порядок ответов:			Да
Нумеровать варианты ответов?			1
Показать количество правильных ответов после окончания:			Да
Штраф за каждую неправильную попытку:			10
ID-номер:			
#	Ответы	Отзыв	Оценка
A.	Списки изменяются динамически (вы можете создать пустой список, потом добавить в него ряд элементов, затем удалить часть из них – и все это будет осуществляться в одном контейнере, относиться к одной и той же переменной);	правильно	16.6
B.	К элементам списка можно получить доступ по индексу (начиная с нулевого). Индексация бывает и обратной, отрицательной (индекс -1 обозначает последний элемент списка, -2 – предпоследний и т.д.);	правильно	16.6
C.	В списках может содержаться ряд элементов любых типов (числа, строки, другие списки ...);	правильно	16.6
D.	У списков имеется большое разнообразие методов, позволяющих осуществлять операции с ними (расширять, удалять элементы, очищать, сортировать);	правильно	16.6
E.	Можно делать «срезы» списков т.е. формировать более мелкие последовательности на основании имеющейся;	правильно	16.6
F.	Значения элементов могут повторяться	правильно	16.6
Общий отзыв к вопросу:			
Для любого правильного ответа:		Ваш ответ верный.	
Для любого неправильного ответа:		Ваш ответ неправильный.	
Для любого частично правильного ответа:		Ваш ответ частично правильный.	
Подсказка 1:			
Показать количество правильных ответов (Подсказка 1):		Нет	
Удалить некорректные ответы (Подсказка 1):		Нет	
Теги:			
<i>Позволяет выбирать один или несколько правильных ответов из заданного списка. (MC/MA)</i>			

Вопрос 14

Какими способами можно получить список ['C', 'т', 'р', 'о', 'к', 'а']			МА
Балл по умолчанию:			1
Случайный порядок ответов:			Да
Нумеровать варианты ответов?			1
Показать количество правильных ответов после окончания:			Да
Штраф за каждую неправильную попытку:			10
ID-номер:			
#	Ответы	Отзыв	Оценка
A.	<p>При помощи функции list():</p> <p>Пример – IDE</p> <pre>text = 'Строка' print(list(text))</pre>	правильно	33.3
B.	<p>При помощи цикла for:</p> <p>Пример – IDE</p> <pre>text = 'Строка' new_list = [] for i in text: new_list.append(i) print(new_list)</pre>	правильно	33.3
C.	<p>При помощи генератора списков:</p> <p>Пример – IDE</p> <pre>text = 'Строка' new_list = [i for i in text] print(new_list)</pre>	правильно	33.3
Общий отзыв к вопросу:			
Для любого правильного ответа:		Ваш ответ верный.	
Для любого неправильного ответа:		Ваш ответ неправильный.	
Для любого частично правильного ответа:		Ваш ответ частично правильный.	
Подсказка 1:			
Показать количество правильных ответов (Подсказка 1):		Нет	
Удалить некорректные ответы (Подсказка 1):		Нет	
Теги:			
Позволяет выбирать один или несколько правильных ответов из заданного списка. (MC/MA)			

Вопрос 14

Установите соответствие: описание действия со списком и код на Python			МАТ
Балл по умолчанию:			1
Перемешать:			Да
Показать количество правильных ответов после окончания:			Да
Штраф за каждую неправильную попытку:			0
ID-номер:			
#	Вопрос	Ответ	
1.	Добавить новый элемент типа str в конец списка	lst.append('Строка')	
2.	Добавить новый элемент типа int на место с индексом	lst[4] = 189	
3.	Добавить новый элемент типа list в конец списка	lst.append(['a', 'b', 'a', 'hello'])	
4.	Добавить новый элемент типа tuple на место с индексом	lst[-3] = (1, 6, 89)	
5.	Получить элемент по индексу	lst[0]	
6.	Удалить элемент	lst.remove(189)	
7.	Найти число повторений элемента списка	lst.count(1)	
8.		1	
	Общий отзыв к вопросу:		
	Для любого правильного ответа:		Ваш ответ верный.
	Для любого неправильного ответа:		Ваш ответ неправильный.
	Для любого частично правильного ответа:		Ваш ответ частично правильный.
	Подсказка 1:		
	Показать количество правильных ответов (Подсказка 1):		Нет
	Удалить некорректные ответы (Подсказка 1):		Нет
	Теги:		
<p><i>Вы должны указать по меньшей мере два вопроса и три ответа. Вы можете включить дополнительные неправильные ответы, создав ответ на пустой вопрос. Записи, где и вопрос и ответ пустые, будут игнорироваться</i></p>			

Вопрос 15

Для выполнения срезов списков установите соответствие программного кода и результата.			МАТ
Балл по умолчанию:			2
Перемешать:			Да
Показать количество правильных ответов после окончания:			Да
Штраф за каждую неправильную попытку:			0
ID-номер:			
#	Вопрос	Ответ	
1.	<code>h = [1, 2, 3, 14, 33, 1, 9]</code> <code>print(h[slice(2, 6, 2)])</code>	[3, 33]	
2.	<code>h = [1, 2, 3, 14, 33, 1, 9]</code> <code>print(h[:,2])</code>	[1, 3, 33, 9]	
3.	<code>h = [1, 2, 3, 14, 33, 1, 9]</code> <code>print(h[2:6:2])</code>	[3, 33]	
4.		[2, 14]	
Общий отзыв к вопросу:			
Для любого правильного ответа:		Ваш ответ верный.	
Для любого неправильного ответа:		Ваш ответ неправильный.	
Для любого частично правильного ответа:		Ваш ответ частично правильный.	
Подсказка 1:			
Показать количество правильных ответов (Подсказка 1):		Нет	
Удалить некорректные ответы (Подсказка 1):		Нет	
Теги:			
<i>Вы должны указать по меньшей мере два вопроса и три ответа. Вы можете включить дополнительные неправильные ответы, создав ответ на пустой вопрос. Записи, где и вопрос и ответ пустые, будут игнорироваться</i>			

Вопрос 16

Для того, чтобы приведенные в коде символы интерпретировались как строка, их нужно выделить особыми символами. Укажите возможные способы.			MA
Балл по умолчанию:			1
Случайный порядок ответов			Да
Нумеровать варианты ответов?			1
Показать количество правильных ответов после окончания:			Да
Штраф за каждую неправильную попытку:			10
ID-номер:			
#	Ответы	Отзыв	Оценка
A.	одинарные кавычки	верно	25
B.	двойные кавычки	верно	25
C.	тройные одинарные кавычки (многострочный текст с сохранением форматирования)	верно	25
D.	тройные двойные кавычки (многострочный текст с сохранением форматирования)	верно	25
E.	знаками &	неверно	0
Общий отзыв к вопросу:			
Для любого правильного ответа:		Ваш ответ верный.	
Для любого неправильного ответа:		Ваш ответ неправильный.	
Для любого частично правильного ответа:		Ваш ответ частично правильный.	
Подсказка 1:			
Показать количество правильных ответов (Подсказка 1):		Нет	
Удалить некорректные ответы (Подсказка 1):		Нет	
Теги:			
<i>Позволяет выбирать один или несколько правильных ответов из заданного списка. (MC/MA)</i>			

Вопрос 17

Укажите соответствие: как применяют операции сложения и умножения к строкам?			МАТ
Балл по умолчанию:			1
Перемешать:			Да
Показать количество правильных ответов после окончания:			Да
Штраф за каждую неправильную попытку:			0
ID-номер:			
#	Вопрос	Ответ	
1.	Операция сложения или конкатенирования	складывает две или более строк в одну. В качестве аргументов могут выступать только строки.	
2.	Операция умножения	применяется к строкам, подразумевает ее дублирование определенное количество раз (в соответствии со значением числа, которое передано в качестве параметра.	
3.	Результат операции умножения	Если передать отрицательное число или ноль, то вернется пустая строка).	
	Общий отзыв к вопросу:		
	Для любого правильного ответа:	Ваш ответ верный.	
	Для любого неправильного ответа:	Ваш ответ неправильный.	
	Для любого частично правильного ответа:	Ваш ответ частично правильный.	
	Подсказка 1:		
	Показать количество правильных ответов (Подсказка 1):	Нет	
	Удалить некорректные ответы (Подсказка 1):	Нет	
	Теги:		
<p><i>Вы должны указать по меньшей мере два вопроса и три ответа. Вы можете включить дополнительные неправильные ответы, создав ответ на пустой вопрос. Записи, где и вопрос и ответ пустые, будут игнорироваться</i></p>			

Вопрос 18

Продемонстрируйте все возможные типовые варианты написания условных инструкций			МА
Балл по умолчанию:			1
Случайный порядок ответов:			Да
Нумеровать варианты ответов?			1
Показать количество правильных ответов после окончания:			Да
Штраф за каждую неправильную попытку:			10
ID-номер:			
#	Ответы	Отзыв	Оценка
A.	Только if	правильно	25
B.	if / else	правильно	25
C.	if / elif	правильно	25
D.	if / elif / else	правильно	25
Общий отзыв к вопросу:			
Для любого правильного ответа:		Ваш ответ верный.	
Для любого неправильного ответа:		Ваш ответ неправильный.	
Для любого частично правильного ответа:		Ваш ответ частично правильный.	
Подсказка 1:			
Показать количество правильных ответов (Подсказка 1):		Нет	
Удалить некорректные ответы (Подсказка 1):		Нет	
Теги:			
<i>Позволяет выбирать один или несколько правильных ответов из заданного списка. (MC/MA)</i>			

Вопрос 19

Перечислите все возможные виды параметров функций			МА
Балл по умолчанию:			1
Случайный порядок ответов			Да
Нумеровать варианты ответов?			1
Показать количество правильных ответов после окончания:			Да
Штраф за каждую неправильную попытку:			10
ID-номер:			
#	Ответы	Отзыв	Оценка
A.	позиционные	верно	25
B.	строго позиционные	верно	25
C.	ключевые	верно	25
D.	строго ключевые	верно	25
E.	временные	неверно	0
Общий отзыв к вопросу:			
Для любого правильного ответа:		Ваш ответ верный.	
Для любого неправильного ответа:		Ваш ответ неправильный.	
Для любого частично правильного ответа:		Ваш ответ частично правильный.	
Подсказка 1:			
Показать количество правильных ответов (Подсказка 1):		Нет	
Удалить некорректные ответы (Подсказка 1):		Нет	
Теги:			
<i>Позволяет выбирать один или несколько правильных ответов из заданного списка. (MC/MA)</i>			

Вопрос 20

Для передачи в функцию строго ключевые аргументы используют инструкции			MA
Балл по умолчанию:			1
Случайный порядок ответов			Да
Нумеровать варианты ответов?			1
Показать количество правильных ответов после окончания:			Да
Штраф за каждую неправильную попытку:			10
ID-номер:			
#	Ответы	Отзыв	Оценка
A.	*	правильно	50
B.	*args	правильно	50
C.	/	неправильно	0
Общий отзыв к вопросу:			
Для любого правильного ответа:		Ваш ответ верный.	
Для любого неправильного ответа:		Ваш ответ неправильный.	
Для любого частично правильного ответа:		Ваш ответ частично правильный.	
Подсказка 1:			
Показать количество правильных ответов (Подсказка 1):		Нет	
Удалить некорректные ответы (Подсказка 1):		Нет	
Теги:			
<i>Позволяет выбирать один или несколько правильных ответов из заданного списка. (MC/MA)</i>			

Вопрос 21

Установите соответствие между назначением и именем функции			МАТ
Балл по умолчанию:			1
Перемешать:			Да
Показать количество правильных ответов после окончания:			Да
Штраф за каждую неправильную попытку:			0
ID-номер:			
#	Вопрос	Ответ	
1.	Возвращает максимальное значение итерабельного объекта (списка, словаря, кортежа...) или переданных аргументов (если их несколько). Возможно использование дополнительных параметров: key (функция предварительной обработки значений объекта), default (значение по умолчанию, если передан пустой объект).	Функция max()	
2.	Определяет количество элементов в объекте (строке, списке, словаре, множестве и т.п.)	Функция len()	
3.	Позволяет узнать уникальный идентификатор любого объекта в рамках конкретной исполняемой среды. При новом запуске скрипта значения идентификаторов поменяются	Функция id()	
4.		Функция lambda()	
	Общий отзыв к вопросу:		
	Для любого правильного ответа:	Ваш ответ верный.	
	Для любого неправильного ответа:	Ваш ответ неправильный.	
	Для любого частично правильного ответа:	Ваш ответ частично правильный.	
	Подсказка 1:		
	Показать количество правильных ответов (Подсказка 1):	Нет	
	Удалить некорректные ответы (Подсказка 1):	Нет	
	Теги:		
<i>Вы должны указать по меньшей мере два вопроса и три ответа. Вы можете включить дополнительные неправильные ответы, создав ответ на пустой вопрос. Записи, где и вопрос и ответ пустые, будут игнорироваться</i>			

Вопрос 22

Установите соответствие: Какой результат дает применение операции <code>int(a)</code> к любому числу «a» с плавающей точкой?			MAT
Балл по умолчанию:			2
Перемешать:			Да
Показать количество правильных ответов после окончания:			Да
Штраф за каждую неправильную попытку:			0
ID-номер:			
#	Вопрос	Ответ	
1.	<code>int(3.93)</code>	3	
2.	<code>int(-11.328)</code>	-11	
3.	<code>int(-5.08)</code>	5	
4.	<code>int(5.68)</code>	-5	
5.		-6	
Общий отзыв к вопросу:			
Для любого правильного ответа:		Ваш ответ верный.	
Для любого неправильного ответа:		Ваш ответ неправильный.	
Для любого частично правильного ответа:		Ваш ответ частично правильный.	
Подсказка 1:			
Показать количество правильных ответов (Подсказка 1):		Нет	
Удалить некорректные ответы (Подсказка 1):		Нет	
Теги:			
<p><i>Вы должны указать по меньшей мере два вопроса и три ответа. Вы можете включить дополнительные неправильные ответы, создав ответ на пустой вопрос. Записи, где и вопрос и ответ пустые, будут игнорироваться</i></p>			

Вопрос 23

Установите соответствие: какие типы данных можно преобразовать в числа с плавающей точкой?		MAT
Балл по умолчанию:		2
Перемешать:		Да
Показать количество правильных ответов после окончания:		Да
Штраф за каждую неправильную попытку:		0
ID-номер:		8.0
#	Вопрос	Ответ
1.	вещественные числа <code>float(8.0)</code>	8.0
2.	булевы значения (т.к. они относятся, в том числе, к типу <code>int</code>) <code>float(False)</code>	0.0
3.	числовые строки <code>float('4.6')</code>	4.6
4.		1
	Общий отзыв к вопросу:	целые числа <code>float(8)</code>
	Для любого правильного ответа:	Ваш ответ верный.
	Для любого неправильного ответа:	Ваш ответ неправильный.
	Для любого частично правильного ответа:	Ваш ответ частично правильный.
	Подсказка 1:	
	Показать количество правильных ответов (Подсказка 1):	Нет
	Удалить некорректные ответы (Подсказка 1):	Нет
	Теги:	
<p><i>Вы должны указать по меньшей мере два вопроса и три ответа. Вы можете включить дополнительные неправильные ответы, создав ответ на пустой вопрос. Записи, где и вопрос и ответ пустые, будут игнорироваться</i></p>		

2. Практические задания с ответами

Задача 1

<p>Напишите функцию <code>pos_add(a, b)</code>, которая возвращает положительное значение сложения двух целых чисел. Выполните 3 разных теста для проверки</p> <p>Рекомендация:</p> <p>Проверку на действительность (целые числа) аргументов не производить, т. к. в условии обозначено, что мы предоставляем только целые числа.</p> <p>Оптимальный вариант – использовать встроенную функцию <code>abs()</code>, которая возвращает положительное число.</p>
<p>Решение</p> <pre>def pos_add(a, b): return abs(a + b) a=int(input('Введите целое a=')) b=int(input('Введите целое b=')) print('Сумма', a,'+',b,'=',pos_add(a, b))</pre>

Задача 2

Для этого нужно написать функцию `first_last(letter, st)`, включающую 2 параметра: `letter` – искомый символ, `st` – целевая строка.

В случае отсутствия буквы в строке, нужно вернуть кортеж `(None, None)`, если же она есть, то кортеж будет состоять из первого и последнего индекса этого символа.

Рекомендации

Решение задачи сводится к оперированию методами `find()` и `rfind()`.

Тесты

```
print(first_last('a', 'abba'))
print(first_last('a', 'abbaaab'))
print(first_last('a', 'a'))
print(first_last('a', 'spring'))
```

Решение

```
def first_last(letter, st):
    first = st.find(letter)
    if first < 0:
        return None, None
    last = st.rfind(letter)
    return first, last
```

Тесты

```
print(first_last('a', 'abba'))
print(first_last('a', 'abbaaab'))
print(first_last('a', 'a'))
print(first_last('a', 'spring'))
```

Результат выполнения

`(0, 3)`

`(0, 6)`

`(0, 0)`

`(None, None)`

Задача 3

Николай написал функцию `is_alive(health)`, которая проверяет здоровье персонажа в игре. Если оно равно или меньше нуля, то функция возвращает `False`, в противном случае `True`. К сожалению, функция не работает, так как ученик допустил в ней ряд ошибок. Исправьте их и проверьте работоспособность программы (в качестве аргумента всегда передается число).

```
def is_alive(health):
    if:
        health < 0
        False
    else:
        return true

# Тесты
print(is_alive(122))
print(is_alive(0))
print(is_alive(-1))
```

Решение

В функции Николая есть ряд ошибок:

- `True` – пишется с большой буквы
- проверка здоровья должна писаться после выражения `if` до двоеточия и включать в себя нулевое значение
- в теле оператора `if` значение `False` должно возвращаться, а не просто объявляться.

Пример – IDE

```
def is_alive(health):
    if health <= 0:
        return False
    else:
        return True
```

Тесты

```
print(is_alive(122))
print(is_alive(0))
print(is_alive(-1))
```

Результат выполнения

```
True
False
False
```

Задача 4

На вход функция `more_than_five(lst)` получает список из целых чисел.

Результатом работы функции должен стать новый список, в котором содержатся только те числа, которые больше 5 по модулю.

Рекомендации

Для решения задачи воспользуйтесь циклом `for`

Тесты

```
print(more_than_five([-11, 4, -2, 90, 400, 0, -5]))
```

```
print(more_than_five([-2, 2, 3, 4, 0, -1]))
```

```
print(more_than_five([70, -900, 41, 0]))
```

Решение

```
def more_than_five(lst):  
    new_lst = []  
    for number in lst:  
        if abs(number) > 5:  
            new_lst.append(number)  
    return new_lst
```

Тесты

```
print(more_than_five([-11, 4, -2, 90, 400, 0, -5]))
```

```
print(more_than_five([-2, 2, 3, 4, 0, -1]))
```

```
print(more_than_five([70, -900, 41, 0]))
```

Результат выполнения

```
[-11, 90, 400]
```

```
[]
```

```
[70, -900, 41]
```

Задача 5

Напишите функцию `sum_range(start, end)`, которая суммирует все целые числа от значения «start» до величины «end» включительно.

Если пользователь задаст первое число большее чем второе, просто поменяйте их местами.

Рекомендации При решении удобно воспользоваться встроенными функциями `range()` и `sum()`.

Тесты

```
print(sum_range(2, 12))
```

```
print(sum_range(-4, 4))
```

```
print(sum_range(3, 2))
```

Решение

```
def sum_range(start, end):  
    if start > end:  
        end, start = start, end  
    return sum(range(start, end + 1))
```

Тесты

```
print(sum_range(2, 12))  
print(sum_range(-4, 4))  
print(sum_range(3, 2))
```

Результат выполнения

77

0

5

Задача 6

На входе функция `to_set()` получает строку или список чисел. Преобразуйте их в множество. На выходе должно получиться множество и его мощность

Рекомендации. Список чисел или строка преобразуются в множество без возможных ошибок. Нужно просто «обернуть» их в конструктор множества и воспользоваться методом [len\(\)](#) для получения ответа.

#тест

```
print(to_set('я обычная строка'))  
print(to_set([4, 5, 4, 6, 2, 9, 11, 3, 4, 2]))
```

Решение

```
def to_set(element):  
    st = set(element)  
    return st, len(st)
```

Тесты

```
print(to_set('я обычная строка'))  
print(to_set([4, 5, 4, 6, 2, 9, 11, 3, 4, 2]))
```

Результат выполнения

({'к', 'б', 'я', 'ы', 'н', 'р', 'т', 'о', 'ч', ' ', 'с', 'а'}, 12)

({2, 3, 4, 5, 6, 9, 11}, 7)

Задача 7

Напишите функцию `to_float(num)`, которая преобразует любое число в число с плавающей точкой.

Если в качестве аргумента передан другой тип данных, она возвращает «Невозможно преобразовать». Выполняйте в режиме терминала (IDLE)

Рекомендации

Необходимо сделать проверку аргумента на принадлежность к типу `int` или `float`. Если это так, то осуществляем преобразование. В противном случае возвращаем «Невозможно преобразовать». Важно учесть, что и булевы значения можно преобразовать в тип `float`.

Тесты:

```
print(to_float(12))
print(to_float(-1.762))
print(to_float(True))
print(to_float('Не число'))
print(to_float('2.2'))
```

Решение

Решение – IDE

```
def to_float(num):
    if isinstance(num, (int, float)):
        return float(num)
    return "Невозможно преобразовать"
```

Тесты

```
print(to_float(12))
print(to_float(-1.762))
print(to_float(True))
print(to_float('Не число'))
print(to_float('2.2'))
```

Результат выполнения

12.0

-1.762

1.0

Невозможно преобразовать

Невозможно преобразовать

Задача 8

Дано 4 числа.

Нужно написать функцию `avg_5(a, b, c, d)`, которая возвращает среднее арифметическое аргументов и округляет его до 5 знаков после запятой. Выполните в режиме терминала (IDLE)

Рекомендации

Для округления воспользуемся встроенной функцией `round()`. Осталось сложить числа и разделить их на 4.

Выполните 3 теста с данными:

1) `a=1, b=6, c=7, d=4`

2) `a=1.7, b=6.2, c=2, d=6`

3) `a=3, b=-3.143223442, c=-4.76, d=1.3902`

Решение

Решение - IDE

```
def avg_5(a, b, c, d):  
    return round((a + b + c + d) / 4, 5)  
  
# Тесты  
print(avg_5(1, 6, 7, 4))  
print(avg_5(1.7, 6.2, 2, 6))  
print(avg_5(3, -3.143223442, -4.76, 1.3902))
```

Результат выполнения

4.5

3.975

-0.87826

Задача 9

Напишите функцию `change(lst)`, которая принимает список и меняет местами его первый и последний элемент. В исходном списке минимум 2 элемента.

Рекомендации

Воспользуйтесь методами `pop()` и `insert()`.

Выполните в режиме терминала (IDLE)

Тесты

```
print(change([1, 2, 3]))
```

```
print(change([1, 2, 3, 4, 5]))
```

```
print(change(['н', 'л', 'о', 'с']))
```


Решение

```
def change(lst):  
    new_start = lst.pop() # Удаляем последний элемент и сохраняем его в переменную  
    new_end = lst.pop(0) # Удаляем первый элемент и сохраняем его в переменную  
    lst.append(new_end) # Добавляем к списку новый последний элемент  
    lst.insert(0, new_start) # Добавляем к списку новый первый элемент  
    return lst
```

Тесты

```
print(change([1, 2, 3]))  
print(change([1, 2, 3, 4, 5]))  
print(change(['н', 'л', 'о', 'с']))
```

Результат выполнения

```
[3, 2, 1]  
[5, 2, 3, 4, 1]  
['с', 'л', 'о', 'н']
```

Задача 10.

Николай задумался о поиске «бесполезного» числа на основании списка.

Суть оного в следующем: он берет произвольный список чисел, находит самое большое из них, а затем делит его на длину списка.

Студент пока не придумал, где может пригодиться подобное значение, но ищет у вас помощи в реализации такой функции `useless(s)`.

Выполните в режиме терминала

```
# Тесты
print(useless([1, 5, 77]))
print(useless([19, 8.3, -4, 11, 0, 5]))
print(useless([-33, -0.05, -4.18, 11.2, 13.12, 55, 7.1]))
```

Решение

```
def useless(lst):
    return max(lst) / len(lst)
# Тесты
print(useless([1, 5, 77]))
print(useless([19, 8.3, -4, 11, 0, 5]))
print(useless([-33, -0.05, -4.18, 11.2, 13.12, 55, 7.1]))
Результат выполнения
25.666666666666668
3.1666666666666665
7.857142857142857
```

Задача 11

Функция `slicer()` на вход принимает кортеж и случайный элемент. Требуется вернуть новый кортеж.

Если элемента нет вовсе – вернуть пустой кортеж.

Если элемент встречается, то определить, сколько раз он встречается и вернуть кортеж, последний элемент которого равен количеству вхождений элемента.

Рекомендации

Используйте особенности кортежа и списка.

#тесты

```
print ("первый пример:",slicer((912,13,77,-3,55), 55))
print ("второй пример:",slicer(("удача",13,77,-3,55), "снег"))
print ("третий пример:",slicer(["удача",2,1],13,77,-3,55), "удача"))
print ("четвертый пример:",slicer(["удача",2,1],13,77,-3,55), ["удача",2,1]))
```

Решение

Неозначанный

```
def slicer(any_tuple, element):
```

```
    if element in any_tuple:
        kol=any_tuple.count(element)
        any_tuple_1=[any_tuple]
        any_tuple_1.append(kol)
        return (any_tuple_1)
    else:
        return ()
```

#тесты

```
print ("первый пример:",slicer((912,13,77,-3,55), 55))
print ("второй пример:",slicer(("удача",13,77,-3,55), "снег"))
print ("третий пример:",slicer(["удача",2,1],13,77,-3,55), "удача"))
print ("четвертый пример:",slicer(["удача",2,1],13,77,-3,55), ["удача",2,1]))
```

решение

```
первый пример: [(912, 13, 77, -3, 55), 1]
второй пример: ()
третий пример: ()
четвертый пример: [('удача', 2, 1), 13, 77, -3, 55), 1]
```

Задача 12

На входе функция `to_set()` получает строку или список чисел. Преобразуйте их в множество. На выходе должно получиться множество и его мощность.

Рекомендации

Список чисел или строка преобразуются в множество без возможных ошибок. Нужно просто обернуть их в конструктор и воспользоваться методом [len\(\)](#) для получения ответа.

Тесты

```
print(to_set('я обычная строка'))
print(to_set([4, 5, 4, 6, 2, 9, 11, 3, 4, 2]))
```

Решение

```
def to_set(element):  
    st = set(element)  
    return st, len(st)
```

Тесты

```
print(to_set('я обычная строка'))
```

```
print(to_set([4, 5, 4, 6, 2, 9, 11, 3, 4, 2]))
```

Результат выполнения

```
({'к', 'б', 'я', 'ы', 'н', 'р', 'т', 'о', 'ч', ' ', 'с', 'а'}, 12)
```

```
({2, 3, 4, 5, 6, 9, 11}, 7)
```

Задача 13

Напишите функцию `superset()`, которая принимает 2 множества. Результат работы функции: вывод в консоль одного из сообщений в зависимости от ситуации:

- 1 - «Супермножество не обнаружено»
- 2 – «Объект {X} является чистым супермножеством»
- 3 – «Множества равны»

Рекомендации

Для написания функции необходимо выявить подчиненность множеств, наличие чистого супермножества.

***Супермножество включает в себя все элементы другого множества и может быть равно с ним по мощности (кол-во элементов в конечном множестве)

При помощи `print()` вывести ответ по результатам оценки. Используйте для вывода f-строки: f-строки берут значения переменных, которые есть в текущей области видимости, и подставляют их в строку. В самой строке лишь нужно указать имя этой переменной в фигурных скобках (например, `print(f"Меня зовут {name} Мне {age} лет.")`)

Тесты

```
set_1 = {1, 8, 3, 5}
set_2 = {3, 5}
set_3 = {5, 3, 8, 1}
set_4 = {90, 100}
superset(set_1, set_2)
superset(set_1, set_3)
superset(set_2, set_3)
superset(set_4, set_2)
```

Решение

```
def superset(set_1, set_2):
    if set_1 > set_2:
        print(f'Объект {set_1} является чистым супермножеством')
    elif set_1 == set_2:
        print(f'Множества равны')
    elif set_1 < set_2:
        print(f'Объект {set_2} является чистым супермножеством')
    else:
        print('Супермножество не обнаружено')
```

Тесты

```
set_1 = {1, 8, 3, 5}
set_2 = {3, 5}
set_3 = {5, 3, 8, 1}
set_4 = {90, 100}
```

```
superset(set_1, set_2)
superset(set_1, set_3)
superset(set_2, set_3)
superset(set_4, set_2)
```

Результат выполнения

```
Объект {8, 1, 3, 5} является чистым супермножеством
Множества равны
Объект {8, 1, 3, 5} является чистым супермножеством
Супермножество не обнаружено
```

Задача 14

Напишите функцию `search_substr(subst, st)`, которая принимает 2 строки и определяет, имеется ли подстрока `subst` в строке `st`.

В случае нахождения подстроки, возвращается фраза «Есть контакт!», а иначе «Мимо!».

Должно быть найдено совпадение независимо от регистра обеих строк.

Рекомендации

Для решения задания необходимо воспользоваться строковыми методами `lower()` и `find()`. Стоит помнить, что `find()` возвращает `-1` в случае ненахождения нужного элемента.

Тесты

```
print(search_substr('Кол', 'коЛокОл'))
print(search_substr('Колобок', 'колобоК'))
print(search_substr('Кол', 'Плов'))
```

Решение

```
def search_substr(subst, st):  
    if subst.lower() in st.lower():  
        return 'Есть контакт!'  
    else:  
        return 'Мимо!'
```

Тесты

```
print(search_substr('Кол', 'коЛокОл'))  
print(search_substr('Колобок', 'колобоК'))  
print(search_substr('Кол', 'Плов'))
```

Результат выполнения

Есть контакт!

Есть контакт!

Мимо!

ПАКЕТ ЭКЗАМЕНАТОРА		
Объекты оценки	Критерии оценки результата	Отметка о выполнении
У-1 Разрабатывать алгоритмы для конкретных задач.	Математическая модель и блок-схема построены правильно, программный код соответствует правилам хорошего стиля программирования, выполнен без ошибок	
У-2 Использовать языки программирования	Программный код соответствует правилам хорошего стиля программирования и требованиям	
У-4 Использовать программы для графического отображения алгоритмов.	Алгоритм задачи построен по правилам формирования блок-схем и/или алгоритмического языка	
У-5 Работать в среде программирования.	Программный код соответствует правилам хорошего стиля программирования	
У-6 Реализовывать построенные алгоритмы в виде программ на конкретном языке программирования.	Программный код соответствует правилам синтаксиса и семантики языка	
У-7 Выполнять проверку, отладку кода программы.	Отсутствие сообщений об ошибках, получение прогнозируемых результатов исполнения программного кода, согласно тестовых наборов	
3-1 Сквозные цифровые технологии и субтехнологии. Этические проблемы цифровизации. Цифровые риски и безопасность.	Приведен полный список сквозных технологий. Перечислены субтехнологии для каждой технологии	
3-2 Понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции.	Алгоритм задачи построен по правилам формирования блок-схем и/или алгоритмического языка	
3-4 Основные элементы языка, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти.	Приведен полный список основных типов данных процедурного языка программирования, программный код содержит операторы и операции, управляющие структуры, структуры данных	
3-7 Объектно-ориентированную модель программирования, основные принципы объектно-ориентированного программирования на примере алгоритмического языка: понятие классов и объектов, их свойств и методов, инкапсуляция и полиморфизма, наследования и	Перечислены точно все основные принципы объектно-ориентированного программирования и дана их краткая характеристика полно и точно определена структура класса, содержащего не менее одного свойства и одного метода; создан объект класса для демонстрации	

переопределения алгоритмические конструкции.	обращения к методу и свойству класса	
--	--------------------------------------	--

8. ПЕРЕЧЕНЬ МАТЕРИАЛОВ, ОБОРУДОВАНИЯ И ИНФОРМАЦИОННЫХ ИСТОЧНИКОВ, ИСПОЛЗУЕМЫХ В АТТЕСТАЦИИ

Компьютерный класс

Специализированная мебель:

Экран настенный – 1 шт.

Компьютерные столы – 22 шт.

Стол письменный – 12 шт.

Кресло компьютерное – 22 шт.

Стулья – 24 шт.

Шкаф для документов – 1 шт.

Технические средства обучения:

Персональные компьютеры (моноблоки) – 24 шт.

Мультимедиа проектор – 1шт.

Аудиоколонки – 1шт.

Информационные средства обучения

Информационная система «Единое окно доступа к образовательным ресурсам. Раздел. Информатика и информационные технологии» [Электронный ресурс]. URL: http://window.edu.ru/catalog/?p_rubr=2.2.75.6

Электронно-библиотечная система IPRbooks [Электронный ресурс]. URL: <http://www.iprbookshop.ru/>

Электронно-библиотечная система Юрайт: <https://urait.ru>

Электронно-библиотечная система Znanium: <https://znanium.com/>

Электронно-библиотечная система Book.ru: <https://book.ru/>

Перечень лицензионного и свободно распространяемого программного обеспечения

1) Антивирусная защита Kaspersky Endpoint Security

2) Astra Linux, Libre Office

3) LMS moodle

Печатные издания:

1. Павловская, Т. А. С/C++. Программирование на языке высокого уровня : учебник / Т. А. Павловская. - Санкт-Петербург : Питер, 2022. - 464 с. - ISBN 978-5-4461-1350-7. - Текст: электронный. - URL: <https://znanium.com/catalog/product/1733748> (дата обращения: 01.04.2024)

2. Немцова, Т. И. Программирование на языке высокого уровня. Программирование на языке С++ : учебное пособие / Т.И. Немцова, С.Ю. Голова, А.И. Терентьев ; под ред. Л.Г. Гагариной. — Москва : ФОРУМ : ИНФРА-М, 2024. — 512 с. + Доп. материалы [Электронный ресурс]. — (Среднее профессиональное образование). - ISBN 978-5-8199-0699-6. - Текст : электронный. - URL: <https://znanium.com/catalog/product/2083383> (дата обращения: 01.04.2024).

3. Х. Дейтел, П. Дейтел. Как программировать на С++. 8-е изд.-М.: Лаборатория знаний, 2022 г.- 1032 с.

Дополнительные источники:

4. Агальцов, В. П. Математические методы в программировании : учебник / В. П. Агальцов. — 2-е изд., перераб. и доп. — Москва : ФОРУМ : ИНФРА-М, 2023. — 240 с. — (Среднее профессиональное образование). - ISBN 978-5-8199-0410-7. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1896458> (дата обращения: 01.04.2024).

3. Огнева, М. В. Программирование на языке C++: практический курс : учебное пособие для среднего профессионального образования / М. В. Огнева, Е. В. Кудрина. — Москва : Издательство Юрайт, 2024. — 335 с. — (Профессиональное образование). — ISBN 978-5-534-05780-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://ezpro.fa.ru:2058/bcode/539772> (дата обращения: 01.04.2024)