

ФЕДЕРАЛЬНОЕ АГЕНТСТВО ЖЕЛЕЗНОДОРОЖНОГО ТРАНСПОРТА

Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Иркутский государственный университет путей сообщения»
(ФГБОУ ВО ИРГУПС)

УТВЕРЖДЕНА
приказом и.о. ректора
от «07» июня 2021 г. № 79

Б1.О.26 Языки программирования

рабочая программа дисциплины

Специальность/направление подготовки – 10.05.03 Информационная безопасность автоматизированных систем

Специализация/профиль – Безопасность открытых информационных систем

Квалификация выпускника – Специалист по защите информации

Форма и срок обучения – очная форма 5 лет, 6 месяцев

Кафедра-разработчик программы – Информационные системы и защита информации

Общая трудоемкость в з.е. – 4
Часов по учебному плану (УП) – 144

Формы промежуточной аттестации
очная форма обучения:
экзамен 4 семестр, курсовая работа 4 семестр

Очная форма обучения

Распределение часов дисциплины по семестрам

Семестр	4	Итого
Вид занятий	Часов по УП	Часов по УП
Аудиторная контактная работа по видам учебных занятий/ в т.ч. в форме ПП*	51	51
– лекции	34	34
– практические (семинарские)		
– лабораторные	17	17
Самостоятельная работа	57	57
Экзамен	36	36
Итого	144	144

ИРКУТСК

Электронный документ выгружен из ЕИС ФГБОУ ВО ИРГУПС и соответствует оригиналу

Подписант ФГБОУ ВО ИРГУПС Трофимов Ю.А.

00a73c5b7b623a969ccad43a81ab346d50 с 08.12.2022 14:32 по 02.03.2024 14:32 GMT+03:00

Подпись соответствует файлу документа



Рабочая программа дисциплины разработана в соответствии с федеральным государственным образовательным стандартом высшего образования – специалитет по специальности 10.05.03 Информационная безопасность автоматизированных систем утвержденным Приказом Минобрнауки России от от 26.11.2020 № 1457.

Программу составил(и):
ст. преподаватель, А.Л. Черепанова

Рабочая программа рассмотрена и одобрена для использования в учебном процессе на заседании кафедры «Информационные системы и защита информации», протокол от «4» июня 2021 г. № 11-2

Зав. кафедрой, к.э.н., доцент

Т.К. Кириллова

1 ЦЕЛИ И ЗАДАЧИ ДИСЦИПЛИНЫ	
1.1 Цели дисциплины	
1	сформировать у обучающихся теоретические знания по основам программных средств системного и прикладного значения;
2	сформировать у обучающихся теоретические и прикладные знания в области программирования на языках высокого уровня для решения профессиональных задач
1.2 Задачи дисциплины	
1	сформировать знания в области парадигм программирования и особенностей языков программирования;
2	сформировать у обучающихся навыки написания программ на языке программирования Python;
3	сформировать у обучающихся навыки эффективного использования возможностей интегрированной среды программирования для написания программ на языке Python
1.3 Цель воспитания и задачи воспитательной работы в рамках дисциплины	
Профессионально-трудовое воспитание обучающихся	
<p>Цель – формирование у обучающихся осознанной профессиональной ориентации, понимания общественного смысла труда и значимости его для себя лично, ответственного, сознательного и творческого отношения к будущей деятельности, профессиональной этики, способности предвидеть изменения, которые могут возникнуть в профессиональной деятельности, и умению работать в изменённых, вновь созданных условиях труда.</p> <p>Цель профессионально-трудового воспитания достигается по мере решения в единстве следующих задач:</p> <ul style="list-style-type: none"> – формирование сознательного отношения к выбранной профессии; – воспитание чести, гордости, любви к профессии, сознательного отношения к профессиональному долгу, понимаемому как личная ответственность и обязанность; – формирование психологии профессионала; – формирование профессиональной культуры, этики профессионального общения; – формирование социальной компетентности и другие задачи, связанные с имиджем профессии и авторитетом транспортной отрасли 	

2 МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП	
Блок/часть ОПОП	Блок 1. Дисциплины / Обязательная часть
2.1 Дисциплины и практики, на которых основывается изучение данной дисциплины	
1	Дисциплина изучается на начальном этапе формирования компетенции
2.2 Дисциплины и практики, для которых изучение данной дисциплины необходимо как предшествующее	
1	Б1.О.28 Технологии и методы программирования
2	Б1.О.39 Программно-аппаратные средства защиты информации
3	Б1.О.40 Разработка и эксплуатация автоматизированных систем в защищенном исполнении
4	Б1.О.53 Теория и практика защиты информации в автоматизированных системах железнодорожного транспорта
5	Б1.О.58 Обработка и анализ больших данных
6	Б3.01(Д) Подготовка к процедуре защиты выпускной квалификационной работы
7	Б3.02(Д) Защита выпускной квалификационной работы

3 ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ, СООТНЕСЕННЫЕ С ТРЕБОВАНИЯМИ К РЕЗУЛЬТАТАМ ОСВОЕНИЯ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ		
Код и наименование компетенции	Код и наименование индикатора достижения компетенции	Планируемые результаты обучения
ОПК-2 Способен применять программные средства системного и прикладного назначения, в том числе отечественного производства, для решения задач профессиональной деятельности	ОПК-2.2 Знает основы программных средств системного и прикладного значения, в том числе отечественного производства	Знать: современные информационные технологии и программные средства, в том числе отечественного производства при решении задач профессиональной деятельности
		Уметь: выбирать современные информационные технологии и программные средства, в том числе отечественного производства при решении задач профессиональной деятельности
		Владеть: навыками применения современных информационных технологий и программных средств, в том числе отечественного производства, при решении задач профессиональной деятельности
ОПК-7 Способен	ОПК-7.1 Знает основные	Знать:

создавать программы на языках общего назначения, применять методы и инструментальные средства программирования для решения профессиональных задач, осуществлять обоснованный выбор инструментария программирования и способов организации программ	языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки программных средств для решения задач в профессиональной деятельности	понятие алгоритма; парадигмы программирования; особенности языков программирования для решения задач профессиональной деятельности
		Уметь: создавать алгоритмы решения задач; реализовать алгоритм решения задач на языке Python с помощью различных видов операторов; отлаживать программу в интегрированной среде разработки (Integrated Development Environment, IDE)
		Владеть: навыками работы в IDE для написания программ на языке Python и отладки программы; навыками работы с управляющими конструкциями и их реализации в Python
	ОПК-7.2 Умеет применять языки программирования и работы с базами данных, современные программные среды разработки программных средств для автоматизации бизнес-процессов, решения прикладных задач различных классов, ведения баз данных и информационных хранилищ	Знать: знать возможности языка Python в решении задач защиты информации; современные интегрированные среды разработки(IDE); лексическую и синтаксическую структуру языка Python; составные элементы Python–программы; понятие переменной, типы данных языка Python и их особенности; базовые алгоритмические конструкции и операторы для их реализации
		Уметь: применять коллекции в решении задач на языке Python
		Владеть: навыками написания программ с использованием коллекций Python
ОПК-7.3 Имеет навыки программирования, отладки и тестирования прототипов программно-технических комплексов задач	Знать: коллекции языка Python; понятие подпрограммы; основные свойства и методы работы с файлами; понятие модульного программирования; виды ошибок в программировании	
	Уметь: создавать программы с использованием функций пользователя; работать с файлами и модулями; обрабатывать исключения в программах	
	Владеть: способами подключения модулей Python; методами разработки модулей пользователя; методами работы с файлами; методами обработки исключений в Python	

4 СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Код	Наименование разделов, тем и видов работ	Семестр	Очная форма				*Код индикатора достижения компетенции
			Часы				
			Лек	Пр	Лаб	СР	
1.0	Раздел 1. Современные информационные технологии и программные средства.						
1.1	Современные информационные технологии и программные средства	4				1	ОПК-2.2
1.2	Алгоритмы. Свойства алгоритмов. Блок-схемы	4				1	ОПК-2.2 ОПК-7.1 ОПК-7.2 ОПК-7.3
1.3	Программные средства для создания алгоритмов в виде блок-схем (MS Visio, онлайн-инструменты)	4				1	ОПК-2.2 ОПК-7.1 ОПК-7.2 ОПК-7.3
2.0	Раздел 2. Языки программирования и интегрированные среды разработки.						
2.1	Парадигмы программирования. Языки программирования. Интегрированные среды разработки (IDE)	4	2			1	ОПК-2.2 ОПК-7.1 ОПК-7.2 ОПК-7.3
3.0	Раздел 3. Программирование на языке Python.						
3.1	Введение в язык программирования Python	4	2				ОПК-2.2 ОПК-7.1 ОПК-7.2 ОПК-7.3
3.2	Условный оператор языка Python. Операторы сравнения. Логический тип данных.	4	2				ОПК-2.2 ОПК-7.1

4 СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Код	Наименование разделов, тем и видов работ	Очная форма				*Код индикатора достижения компетенции	
		Семестр	Часы				
			Лек	Пр	Лаб		СР
						ОПК-7.2 ОПК-7.3	
3.3	Решение задач линейной и разветвляющейся структуры	4			2	2	ОПК-2.2 ОПК-7.1 ОПК-7.2 ОПК-7.3
3.4	Операторы цикла в языке Python	4	2				ОПК-2.2 ОПК-7.1 ОПК-7.2 ОПК-7.3
3.5	Коллекции Python. Последовательности. Виды копирования.	4	4				ОПК-2.2 ОПК-7.1 ОПК-7.2 ОПК-7.3
3.6	Решение задач циклической структуры	4			2	3	ОПК-2.2 ОПК-7.1 ОПК-7.2 ОПК-7.3
3.7	Массивы (списки) на языке Python	4	2				ОПК-7.2 ОПК-7.3
3.8	Множества в языке Python	4	2			2	ОПК-2.2 ОПК-7.1 ОПК-7.2 ОПК-7.3
3.9	Решение задач с использованием коллекций Python	4			4	3	ОПК-2.2 ОПК-7.1 ОПК-7.2 ОПК-7.3
3.10	Решение задач на массивы (списки)	4			2	4	ОПК-2.2 ОПК-7.1 ОПК-7.2 ОПК-7.3
3.11	Функции пользователя в Python. Рекурсия.	4	4		3	3	ОПК-2.2 ОПК-7.1 ОПК-7.2 ОПК-7.3
3.12	Функции map, filter, zip, reduce	4	2			2	ОПК-2.2 ОПК-7.1 ОПК-7.2 ОПК-7.3
3.13	Работа с файлами и каталогами в Python	4	2		2	2	ОПК-2.2 ОПК-7.1 ОПК-7.2 ОПК-7.3
3.14	Модули и пакеты в Python	4	2			2	ОПК-2.2 ОПК-7.1 ОПК-7.2 ОПК-7.3
3.15	Ошибки и исключения в Python	4	2		2	2	ОПК-2.2 ОПК-7.1 ОПК-7.2 ОПК-7.3
3.16	Основы объектно-ориентированного программирования в Python	4	4			2	ОПК-2.2 ОПК-7.1 ОПК-7.2 ОПК-7.3
3.17	Создание приложений с графическим интерфейсом	4	2			2	ОПК-2.2 ОПК-7.1 ОПК-7.2 ОПК-7.3

4 СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Код	Наименование разделов, тем и видов работ	Семестр	Очная форма				*Код индикатора достижения компетенции
			Часы				
			Лек	Пр	Лаб	СР	
3.18	Подготовка к тестированию	4				3	ОПК-2.2 ОПК-7.1 ОПК-7.2 ОПК-7.3
	Форма промежуточной аттестации – экзамен	4	36				ОПК-2.2 ОПК-7.1 ОПК-7.2 ОПК-7.3
	Курсовая работа	4				21	ОПК-2.2 ОПК-7.1 ОПК-7.2 ОПК-7.3
	Итого часов (без учёта часов на промежуточную аттестацию)		34		17	57	

5 ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ПРОВЕДЕНИЯ ТЕКУЩЕГО КОНТРОЛЯ УСПЕВАЕМОСТИ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ

Фонд оценочных средств для проведения текущего контроля успеваемости и промежуточной аттестации по дисциплине оформлен в виде приложения № 1 к рабочей программе дисциплины и размещен в электронной информационно-образовательной среде Университета, доступной обучающемуся через его личный кабинет

6 УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

6.1 Учебная литература

6.1.1 Основная литература

	Библиографическое описание	Кол-во экз. в библиотеке/ онлайн
6.1.1.1	Городня, Л. В. Парадигма программирования : учебное пособие для вузов - 2-е изд., стер. / Л. В. Городня. Санкт-Петербург : Лань, 2021. - 232с. - Текст: электронный. - URL: https://e.lanbook.com/book/151660 (дата обращения: 19.04.2023)	Онлайн
6.1.1.2	Шелудько, В. М. Основы программирования на языке высокого уровня Python : учебное пособие / В. М. Шелудько. Ростов-на-Дону, Таганрог : Южный федеральный университет, 2017. - 147с. - Текст: электронный. - URL: https://biblioclub.ru/index.php?page=book&id=500056 (дата обращения: 14.09.2022)	Онлайн
6.1.1.3	Шелудько, В. М. Язык программирования высокого уровня Python: функции, структуры данных, дополнительные модули : учебное пособие / В. М. Шелудько. Ростов-на-Дону, Таганрог : Южный федеральный университет, 2017. - 108с. - Текст: электронный. - URL: https://biblioclub.ru/index.php?page=book&id=500060 (дата обращения: 14.09.2022)	Онлайн

6.1.2 Дополнительная литература

	Библиографическое описание	Кол-во экз. в библиотеке/ онлайн
6.1.2.1	Мирошниченко, И. И. Языки и методы программирования : учебное пособие / И. И. Мирошниченко, Е. Г. Веретенникова, Н. Г. Савельева. Ростов-на-Дону : Издательско-полиграфический комплекс РГЭУ (РИНХ), 2019. - 188с. - Текст: электронный. - URL: https://biblioclub.ru/index.php?page=book&id=567706 (дата обращения: 14.09.2022)	Онлайн

6.1.3 Учебно-методические разработки (в т. ч. для самостоятельной работы обучающихся)

	Библиографическое описание	Кол-во экз. в библиотеке/ онлайн
6.1.3.1	Черепанова, А.Л. Методические указания по изучению дисциплины Б1.О.26	Онлайн

	Языки программирования по специальности 10.05.03 Информационная безопасность автоматизированных систем, специализация Безопасность открытых информационных систем / А.Л. Черепанова ; ИрГУПС. – Иркутск : ИрГУПС, 2023. – 14 с. - Текст: электронный. - URL: https://www.irgups.ru/eis/for_site/umkd_files/mu_7683_1529_2021_1_signed.pdf
6.2 Ресурсы информационно-телекоммуникационной сети «Интернет»	
6.3 Программное обеспечение и информационные справочные системы	
6.3.1 Базовое программное обеспечение	
6.3.1.1	Microsoft Windows Professional 10, государственный контракт от 20.07.2021 № 0334100010021000013-01
6.3.1.2	Microsoft Office Russian 2010, государственный контракт от 20.07.2021 № 0334100010021000013-01
6.3.1.3	FoxitReader, свободно распространяемое программное обеспечение http://free-software.com.ua/pdf-viewer/foxit-reader/
6.3.1.4	Adobe Acrobat Reader DC свободно распространяемое программное обеспечение https://get.adobe.com/ru/reader/enterprise/
6.3.1.5	Яндекс. Браузер. Прикладное программное обеспечение общего назначения, Офисные приложения, лицензия – свободно распространяемое программное обеспечение по лицензии BSD License
6.3.2 Специализированное программное обеспечение	
6.3.2.1	MathCAD_student 15.0 Academic License, Customer Number 434692, контракт от 03.12.2012 № 0334100010012000148-0000756-01
6.3.2.2	Python 3.9, свободно распространяемое программное обеспечение https://docs.python.org/3/license.html
6.3.2.3	Dev-C++, свободная интегрированная среда разработки приложений для языков программирования C/C++, https://code-live.ru/post/dev-cpp-free-cpp-ide-for-windows/
6.3.2.4	MatLab Classroom, R2015a, R2015b, контракт от 09.07.2014 № 0334100010014000028-0000756-01.
6.3.2.5	MatLab Classroom, R2010a, R2010b, лицензия от 16.03.2011 № 689810, ГК № 0334100010011000032-00000756-01.
6.3.3 Информационные справочные системы	
6.3.3.1	Не предусмотрены
6.4 Правовые и нормативные документы	
6.4.1	Не предусмотрены

7 ОПИСАНИЕ МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЙ БАЗЫ, НЕОБХОДИМОЙ ДЛЯ ОСУЩЕСТВЛЕНИЯ УЧЕБНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ	
1	Корпуса А, Б, В, Г, Д, Е ИрГУПС находятся по адресу г. Иркутск, ул. Чернышевского, д. 15; корпус Л ИрГУПС находится – по адресу г. Иркутск, ул. Лермонтова, д.80
2	Учебная аудитория Д-417 для проведения лекционных и практических занятий, лабораторных работ, групповых и индивидуальных консультаций, курсового проектирования (выполнения курсовых работ), текущего контроля и промежуточной аттестации. Основное оборудование: специализированная мебель, мультимедиапроектор, экран, (ноутбук переносной)
3	Учебная аудитория Д-521 для проведения лекционных и практических занятий, лабораторных работ, групповых и индивидуальных консультаций, курсового проектирования (выполнения курсовых работ), текущего контроля и промежуточной аттестации. Основное оборудование: специализированная мебель, мультимедиапроектор, экран, (ноутбук переносной)
4	Лаборатория Д-514 «Проектирование и эксплуатация программно-информационных систем» для проведения лекционных и практических занятий, лабораторных работ, групповых и индивидуальных консультаций, курсового проектирования (выполнения курсовых работ), текущего контроля и промежуточной аттестации. Основное оборудование: специализированная мебель, мультимедиапроектор(переносной),экран(переносной),компьютер
5	Помещения для самостоятельной работы обучающихся оснащены компьютерной техникой, подключенной к информационно-телекоммуникационной сети «Интернет», и обеспечены доступом в электронную информационно-образовательную среду ИрГУПС. Помещения для самостоятельной работы обучающихся: – читальные залы; – учебные залы вычислительной техники А-401, А-509, А-513, А-516, Д-501, Д-503, Д-505, Д-507; – помещения для хранения и профилактического обслуживания учебного оборудования – А-521

8 МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ	
Вид учебной	Организация учебной деятельности обучающегося

деятельности	
Лекция	<p>Лекция (от латинского «lectio» – чтение) – вид аудиторных учебных занятий. Лекция: закладывает основы научных знаний в систематизированной, последовательной, обобщенной форме; раскрывает состояние и перспективы развития соответствующей области науки и техники; концентрирует внимание обучающихся на наиболее сложных, узловых вопросах; стимулирует познавательную активность обучающихся.</p> <p>Во время лекционных занятий обучающийся должен уметь сконцентрировать внимание на изучаемых проблемах и включить в работу все виды памяти: словесную, образную и моторно-двигательную. Для этого весь материал, излагаемый преподавателем, обучающемуся необходимо конспектировать. На полях конспекта следует пометить вопросы, выделенные обучающимся для консультации с преподавателем. Выводы, полученные в виде формул, рекомендуется в конспекте подчеркивать или обводить рамкой, чтобы лучше запоминались. Полезно составить краткий справочник, содержащий определения важнейших понятий лекции. К каждому занятию следует разобрать материал предыдущей лекции. Изучая материал по учебнику или конспекту лекций, следует переходить к следующему вопросу только в том случае, когда хорошо усвоен предыдущий вопрос. Ряд вопросов дисциплины может быть вынесен на самостоятельное изучение. Такое задание требует оперативного выполнения. В конспекте лекций необходимо оставить место для освещения упомянутых вопросов. Обозначить вопросы, термины, материал, который вызывает трудности, пометить и попытаться найти ответ в рекомендуемой литературе. Если самостоятельно не удастся разобраться в материале, то необходимо сформулировать вопрос и задать преподавателю на консультации, на практическом занятии</p>
Практическое занятие	<p>Практическое занятие – вид аудиторных учебных занятий, целенаправленная форма организации учебного процесса, при реализации которой обучающиеся по заданию и под руководством преподавателя выполняют практические задания. Практические задания направлены на углубление научно-теоретических знаний и овладение определенными методами работы, в процессе которых вырабатываются умения и навыки выполнения тех или иных учебных действий в данной сфере науки. Практические занятия развивают научное мышление и речь, позволяют проверить знания обучающихся, выступают как средства оперативной обратной связи; цель практических занятий – углублять, расширять, детализировать знания, полученные на лекции, в обобщенной форме и содействовать выработке навыков профессиональной деятельности.</p> <p>На практических занятиях подробно рассматриваются основные вопросы дисциплины, разбираются основные типы задач. К каждому практическому занятию следует заранее самостоятельно выполнить домашнее задание и выучить лекционный материал к следующей теме. Систематическое выполнение домашних заданий обязательно и является важным фактором, способствующим успешному усвоению дисциплины</p>
Лабораторная работа	<p>Основной целью лабораторных работ является теоретическое обоснование, наглядное и/или экспериментальное подтверждение и/или проверка существенных теоретических положений (законов, закономерностей) анализ существующих методик и методов их реализации и т.д. Они занимают преимущественное место при изучении дисциплин обязательной части и части, формируемой участниками образовательных отношений Блока I.</p> <p>Исходя из цели, содержанием лабораторных работ могут быть:</p> <ul style="list-style-type: none"> - экспериментальная проверка формул, методик расчета; - проведение натурных измерений свойств, рабочих параметров, режимов работы при помощи лабораторного оборудования и/или стендов и макетов; - ознакомление, анализ и теоретические выкладки по устройству, принципу действия и способам обслуживания аппаратов, деталей машин, механизмов, процессов, протекающих в них при этом и т.д.; - наглядная графическая интерпретация чертежей, схем, объемных поверхностей и т.д., воспроизводимых с помощью специализированного программного обеспечения; - имитационное моделирование процессов, протекающих в сложных химических, физических, механических, электрических и пр. объектах; - наглядное представление о работе персонала конкретной организации или подразделения ОАО «РЖД» посредством моделирования штатных и внештатных ситуаций в виртуальных специализированных АРМ (автоматизированных рабочих мест); - установление и подтверждение закономерностей (путем сравнения проведенного эксперимента и рассчитанных значений) и т.д.; - ознакомление с методиками проведения экспериментов, наглядным устройством стенд-макетов и пр.; - установление свойств веществ, их качественных и количественных характеристик; - анализ различных характеристик процессов, в том числе производственных и иных процессов;

	<ul style="list-style-type: none"> - расчет параметров различных явлений и процессов, смоделировать которые не возможно в реальных условиях (например, чрезвычайные ситуации и пр.); - наблюдение развития явлений, процессов и др. <p>Допускается иное содержание лабораторных работ, если это будет способствовать реализации целей и задач дисциплины и формированию соответствующих компетенций.</p> <p>По характеру выполняемых лабораторных работ возможны:</p> <ul style="list-style-type: none"> - ознакомительные работы, используемые для закрепления изученного теоретического материалы; - аналитические работы, используемые для получения новой информации на основе формализованных методов; - творческие работы, ориентированные на самостоятельный выбор подходов решения задач. <p>Прежде, чем приступить к лабораторным занятиям, обучающимся необходимо повторить теоретический материал по теме работы. Каждая лабораторная работа оснащена методическими указаниями, разработанными преподавателями, ведущими дисциплину</p>
Самостоятельная работа	<p>Обучение по дисциплине «Языки программирования» предусматривает активную самостоятельную работу обучающегося. В разделе 4 рабочей программы, который называется «Структура и содержание дисциплины», все часы самостоятельной работы расписаны по темам и вопросам, а также указана необходимая учебная литература: обучающийся изучает учебный материал, разбирает примеры и решает разноуровневые задачи в рамках выполнения как общих домашних заданий, так и индивидуальных домашних заданий (ИДЗ) и других видов работ, предусмотренных рабочей программой дисциплины. При выполнении домашних заданий обучающемуся следует обратиться к задачам, решенным на предыдущих практических занятиях, решенным домашним работам, а также к примерам, приводимым лектором. Если этого будет недостаточно для выполнения всей работы можно дополнительно воспользоваться учебными пособиями, приведенными в разделе 6.1 «Учебная литература». Если, несмотря на изученный материал, задание выполнить не удастся, то в обязательном порядке необходимо посетить консультацию преподавателя, ведущего практические занятия, и/или консультацию лектора.</p> <p>Домашние задания, индивидуальные домашние задания и другие работы, предусмотренные рабочей программой дисциплины должны быть выполнены обучающимся в установленные преподавателем сроки в соответствии с требованиями к оформлению текстовой и графической документации, сформулированным в Положении «Требования к оформлению текстовой и графической документации. Нормоконтроль»</p>
Комплекс учебно-методических материалов по всем видам учебной деятельности, предусмотренным рабочей программой дисциплины (модуля), размещен в электронной информационно-образовательной среде ИрГУПС, доступной обучающемуся через его личный кабинет	

Приложение № 1 к рабочей программе

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

**для проведения текущего контроля успеваемости
и промежуточной аттестации**

1. Общие положения

Фонд оценочных средств (ФОС) является составной частью нормативно-методического обеспечения системы оценки качества освоения обучающимися образовательной программы.

Фонд оценочных средств предназначен для использования обучающимися, преподавателями, администрацией ИрГУПС, а также сторонними образовательными организациями для оценивания качества освоения образовательной программы и уровня сформированности компетенций у обучающихся.

Задачами ФОС являются:

- оценка достижений обучающихся в процессе изучения дисциплины;
- обеспечение соответствия результатов обучения задачам будущей профессиональной деятельности через совершенствование традиционных и внедрение инновационных методов обучения в образовательный процесс;
- самоподготовка и самоконтроль обучающихся в процессе обучения.

Фонд оценочных средств сформирован на основе ключевых принципов оценивания: валидность, надежность, объективность, эффективность.

Для оценки уровня сформированности компетенций используется трехуровневая система:

- минимальный уровень освоения, обязательный для всех обучающихся по завершению освоения образовательной программы; дает общее представление о виде деятельности, основных закономерностях функционирования объектов профессиональной деятельности, методов и алгоритмов решения практических задач;
- базовый уровень освоения, превышение минимальных характеристик сформированности компетенций; позволяет решать типовые задачи, принимать профессиональные и управленческие решения по известным алгоритмам, правилам и методикам;
- высокий уровень освоения, максимально возможная выраженность характеристик компетенций; предполагает готовность решать практические задачи повышенной сложности, нетиповые задачи, принимать профессиональные и управленческие решения в условиях неполной определенности, при недостаточном документальном, нормативном и методическом обеспечении.

2. Перечень компетенций, в формировании которых участвует дисциплина.

Программа контрольно-оценочных мероприятий. Показатели оценивания компетенций, критерии оценки

Дисциплина «Языки программирования» участвует в формировании компетенций:

ОПК-2. Способен применять программные средства системного и прикладного назначений, в том числе отечественного производства, для решения задач профессиональной деятельности

ОПК-7. Способен создавать программы на языках общего назначения, применять методы и инструментальные средства программирования для решения профессиональных задач, осуществлять обоснованный выбор инструментария программирования и способов организации программ

Программа контрольно-оценочных мероприятий очная форма обучения

№	Наименование контрольно-оценочного мероприятия	Объект контроля	Код индикатора достижения компетенции	Наименование оценочного средства (форма проведения*)
4 семестр				
1.0	Раздел 1. Современные информационные технологии и программные средства			
2.0	Раздел 2. Языки программирования и интегрированные среды разработки			
3.0	Раздел 3. Программирование на языке Python			
3.1	Текущий контроль	Решение задач линейной и разветвляющейся структуры	ОПК-2.2 ОПК-7.1 ОПК-7.2 ОПК-7.3	Лабораторная работа (письменно/устно)
3.2	Текущий контроль	Решение задач циклической структуры	ОПК-2.2 ОПК-7.1 ОПК-7.2 ОПК-7.3	Лабораторная работа (письменно/устно)
3.3	Текущий контроль	Решение задач с использованием коллекций Python	ОПК-2.2 ОПК-7.1 ОПК-7.2 ОПК-7.3	Лабораторная работа (письменно/устно)
3.4	Текущий контроль	Решение задач на массивы (списки)	ОПК-2.2 ОПК-7.1 ОПК-7.2 ОПК-7.3	Лабораторная работа (письменно/устно)
3.5	Текущий контроль	Функции пользователя в Python. Рекурсия.	ОПК-2.2 ОПК-7.1 ОПК-7.2 ОПК-7.3	Лабораторная работа (письменно/устно)
3.6	Текущий контроль	Работа с файлами и каталогами в Python	ОПК-2.2 ОПК-7.1 ОПК-7.2 ОПК-7.3	Лабораторная работа (письменно/устно)
3.7	Текущий контроль	Ошибки и исключения в Python	ОПК-2.2 ОПК-7.1 ОПК-7.2 ОПК-7.3	Лабораторная работа (письменно/устно)
	Промежуточная аттестация	Раздел 1. Современные информационные технологии и программные средства. Раздел 2. Языки программирования и интегрированные среды разработки. Раздел 3. Программирование на языке Python.	ОПК-2.2 ОПК-7.1 ОПК-7.2 ОПК-7.3	Курсовая работа (письменно) Курсовая работа (устно)
	Промежуточная аттестация	Раздел 1. Современные информационные технологии и	ОПК-2.2 ОПК-7.1	Экзамен (собеседование) Экзамен - тестирование

		программные средства. Раздел 2. Языки программирования и интегрированные среды разработки. Раздел 3. Программирование на языке Python.	ОПК-7.2 ОПК-7.3	(компьютерные технологии)
--	--	--	--------------------	---------------------------

*Форма проведения контрольно-оценочного мероприятия: устно, письменно, компьютерные технологии.

Описание показателей и критериев оценивания компетенций.

Описание шкал оценивания

Контроль качества освоения дисциплины включает в себя текущий контроль успеваемости и промежуточную аттестацию. Текущий контроль успеваемости и промежуточная аттестация обучающихся проводятся в целях установления соответствия достижений обучающихся поэтапным требованиям образовательной программы к результатам обучения и формирования компетенций.

Текущий контроль успеваемости – основной вид систематической проверки знаний, умений, навыков обучающихся. Задача текущего контроля – оперативное и регулярное управление учебной деятельностью обучающихся на основе обратной связи и корректировки. Результаты оценивания учитываются в виде средней оценки при проведении промежуточной аттестации.

Для оценивания результатов обучения используется четырехбалльная шкала: «отлично», «хорошо», «удовлетворительно», «неудовлетворительно» и двухбалльная шкала: «зачтено», «не зачтено».

Перечень оценочных средств, используемых для оценивания компетенций, а также краткая характеристика этих средств приведены в таблице.

Текущий контроль

№	Наименование оценочного средства	Краткая характеристика оценочного средства	Представление оценочного средства в ФОС
1	Лабораторная работа	Средство, позволяющее оценить умение обучающегося письменно/устно излагать суть поставленной задачи, самостоятельно применять стандартные методы решения поставленной задачи с использованием имеющейся лабораторной базы, проводить анализ полученного результата работы. Может быть использовано для оценки умений, навыков и (или) опыта деятельности обучающихся	Образец задания для выполнения лабораторной работы и примерный перечень вопросов для ее защиты

Промежуточная аттестация

№	Наименование оценочного средства	Краткая характеристика оценочного средства	Представление оценочного средства в ФОС
1	Экзамен	Средство, позволяющее оценить знания, умения, навыков и (или) опыта деятельности обучающегося по дисциплине. Может быть использовано для оценки знаний, умений, навыков и (или) опыта деятельности обучающихся	Перечень теоретических вопросов и практических заданий (образец экзаменационного билета) к экзамену
2	Тест – промежуточная аттестация в форме экзамена	Система автоматизированного контроля освоения компетенций (части компетенций) обучающимся по дисциплине (модулю) с использованием информационно-коммуникационных технологий. Может быть использовано для оценки знаний, умений, навыков и (или) опыта деятельности обучающихся	Фонд тестовых заданий

3	Курсовая работа	<p>Позволяет оценить умения обучающихся самостоятельно конструировать свои знания в процессе решения практических задач и проблем, ориентироваться в информационном пространстве и уровень сформированности аналитических, исследовательских навыков, навыков практического и творческого мышления. Может выполняться в индивидуальном порядке или группой обучающихся.</p> <p>Может быть использовано для оценки умений, навыков и (или) опыта деятельности обучающихся в предметной или межпредметной областях</p>	Образец задания для выполнения курсовой работы и примерный перечень вопросов для ее защиты
---	-----------------	--	--

Критерии и шкалы оценивания компетенций в результате изучения дисциплины при проведении промежуточной аттестации в форме экзамена. Шкала оценивания уровня освоения компетенций

Шкала оценивания	Критерии оценивания	Уровень освоения компетенции
«отлично»	Обучающийся правильно ответил на теоретические вопросы. Показал отличные знания в рамках учебного материала. Правильно выполнил практические задания. Показал отличные умения и владения навыками применения полученных знаний и умений при решении задач в рамках учебного материала. Ответил на все дополнительные вопросы	Высокий
«хорошо»	Обучающийся с небольшими неточностями ответил на теоретические вопросы. Показал хорошие знания в рамках учебного материала. С небольшими неточностями выполнил практические задания. Показал хорошие умения и владения навыками применения полученных знаний и умений при решении задач в рамках учебного материала. Ответил на большинство дополнительных вопросов	Базовый
«удовлетворительно»	Обучающийся с существенными неточностями ответил на теоретические вопросы. Показал удовлетворительные знания в рамках учебного материала. С существенными неточностями выполнил практические задания. Показал удовлетворительные умения и владения навыками применения полученных знаний и умений при решении задач в рамках учебного материала. Допустил много неточностей при ответе на дополнительные вопросы	Минимальный
«неудовлетворительно»	Обучающийся при ответе на теоретические вопросы и при выполнении практических заданий продемонстрировал недостаточный уровень знаний и умений при решении задач в рамках учебного материала. При ответах на дополнительные вопросы было допущено множество неправильных ответов	Компетенция не сформирована

Тест – промежуточная аттестация в форме экзамена

Критерии оценивания	Шкала оценивания
Обучающийся верно ответил на 90 – 100 % тестовых заданий при прохождении тестирования	«отлично»
Обучающийся верно ответил на 80 – 89 % тестовых заданий при прохождении тестирования	«хорошо»
Обучающийся верно ответил на 70 – 79 % тестовых заданий при прохождении тестирования	«удовлетворительно»
Обучающийся верно ответил на 69 % и менее тестовых заданий при прохождении тестирования	«неудовлетворительно»

Курсовая работа

Шкала оценивания	Критерии оценивания
------------------	---------------------

«отлично»	Содержание курсовой работы полностью соответствует заданию. Представлены результаты обзора литературных и иных источников. Структура курсовой работы логически и методически выдержана. Все выводы и предложения убедительно аргументированы. Оформление курсовой работы и полученные результаты полностью отвечают требованиям, изложенным в методических указаниях. При защите курсовой работы обучающийся правильно и уверенно отвечает на вопросы преподавателя, демонстрирует глубокое знание теоретического материала, способен аргументировать собственные утверждения и выводы
«хорошо»	Содержание курсовой работы полностью соответствует заданию. Представлены результаты обзора литературных и иных источников. Структура курсовой работы логически и методически выдержана. Большинство выводов и предложений аргументировано. Оформление курсовой работы и полученные результаты в целом отвечают требованиям, изложенным в методических указаниях. Имеются одна-две несущественные ошибки в использовании терминов, в построенных диаграммах и схемах. Наличествует незначительное количество грамматических и/или стилистических ошибок. При защите курсовой работы обучающийся правильно и уверенно отвечает на большинство вопросов преподавателя, демонстрирует хорошее знание теоретического материала, но не всегда способен аргументировать собственные утверждения и выводы. При наводящих вопросах преподавателя исправляет ошибки в ответе
«удовлетворительно»	Содержание курсовой работы частично не соответствует заданию. Результаты обзора литературных и иных источников представлены недостаточно полно. Есть нарушения в логике изложения материала. Аргументация выводов и предложений слабая или отсутствует. Имеются одно-два существенных отклонений от требований в оформлении курсовой работы. Полученные результаты в целом отвечают требованиям, изложенным в методических указаниях. Имеются одна-две существенных ошибки в использовании терминов, в построенных диаграммах и схемах. Много грамматических и/или стилистических ошибок. При защите курсовой работы обучающийся допускает грубые ошибки при ответах на вопросы преподавателя и /или не дал ответ более чем на 30% вопросов, демонстрирует слабое знание теоретического материала, в большинстве случаев не способен уверенно аргументировать собственные утверждения и выводы
«неудовлетворительно»	Содержание курсовой работы в целом не соответствует заданию. Имеются более двух существенных отклонений от требований в оформлении курсовой работы. Большое количество существенных ошибок по сути работы, много грамматических и стилистических ошибок и др. Полученные результаты не отвечают требованиям, изложенным в методических указаниях. При защите курсовой работы обучающийся демонстрирует слабое понимание программного материала. Курсовая работа не представлена преподавателю. Обучающийся не явился на защиту курсовой работы

Критерии и шкалы оценивания результатов обучения при проведении текущего контроля успеваемости

Лабораторная работа

Шкалы оценивания		Критерии оценивания
«отлично»	«зачтено»	Лабораторная работа выполнена в обозначенный преподавателем срок, письменный отчет без замечаний. Лабораторная работа выполнена обучающимся в полном объеме с соблюдением необходимой последовательности. Обучающийся работал полностью самостоятельно; показал необходимые для проведения работы теоретические знания, практические умения и навыки. Работа (отчет) оформлена аккуратно, в наиболее оптимальной для фиксации результатов форме
«хорошо»		Лабораторная работа выполнена в обозначенный преподавателем срок, письменный отчет с небольшими недочетами. Лабораторная работа выполнена обучающимся в полном объеме и самостоятельно. Допущены отклонения от необходимой последовательности выполнения, не влияющие на правильность конечного результата. Работа показывает знание обучающимся основного теоретического материала и овладение умениями, необходимыми для самостоятельного выполнения работы. Допущены неточности и небрежность в оформлении результатов работы (отчета)

«удовлетворительно»		Лабораторная работа выполнена с задержкой, письменный отчет с недочетами. Лабораторная работа выполняется и оформляется обучающимся при посторонней помощи. На выполнение работы затрачивается много времени. Обучающийся показывает знания теоретического материала, но испытывает затруднение при самостоятельной работе с источниками знаний или приборами
«неудовлетворительно»	«не зачтено»	Лабораторная работа не выполнена, письменный отчет не представлен. Результаты, полученные обучающимся, не позволяют сделать правильных выводов и полностью расходятся с поставленной целью. Показывается плохое знание теоретического материала и отсутствие необходимых умений. Лабораторная работа не выполнена, у учащегося отсутствуют необходимые для проведения работы теоретические знания, практические умения и навыки

3. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности

3.1 Типовые задания для выполнения лабораторной работы и примерный перечень вопросов для ее защиты

Контрольные варианты заданий выложены в электронной информационно-образовательной среде ИрГУПС, доступной обучающемуся через его личный кабинет.

Ниже приведен образец типовых вариантов заданий для выполнения лабораторных работ и примерный перечень вопросов для их защиты.

Образец заданий для выполнения лабораторных работ и примерный перечень вопросов для их защиты

«Решение задач линейной и разветвляющейся структуры»

Задание: Составить программу по варианту

С клавиатуры вводится четырёхзначное число. Найти

- сумму первой и последней цифры числа;
- произведение второй и третьей цифры числа.

Вычислить значение функции

$$t = \frac{2 \cos(x - \pi/6)}{0,5 + \sin^2 y} \left(1 + \frac{z^2}{3 - z^2/5} \right).$$

При $x = 14.26$, $y = -1.22$,

$$z = 3.5 \times 10^{-2} \quad ; \quad \mathbf{0.564846}.$$

Шахматная доска содержит 8×8 клеток. Горизонтالي и вертикали обозначаются цифрами от 1 – 8. Написать программу, проверяющую, можно ли с первой заданной клетки попасть на вторую одним ходом ладьи.

Вычислить значение Y в зависимости от выбранной функции $\varphi(x)$, аргумент которой определяется из поставленного условия. Возможные значения функции $\varphi(x)$: $2x$, x^3 , $x/2$. Пользователь должен указать при какой функции $\varphi(x)$ программа будет выполнять вычисления. Предусмотреть вывод сообщений, показывающих, при каком условии и с какой функцией производились вычисления Y .

Предусмотреть диалог с пользователем и дружественный интерфейс.

$$y = 2a \cos^3 x^2 + \sin^2 x^3 - b\varphi(x), \quad \text{где } x = \begin{cases} z^3 + 0,2; & z < 1; \\ z + \ln z; & z \geq 1. \end{cases}$$

Составить блок-схему к программе.

Контрольные вопросы:

- 1) Опишите основные характеристики языка Python
- 2) Что такое трансляторы, компиляторы, интерпретаторы?
- 3) Особенности синтаксиса языка Python
- 4) Стилль наименования идентификаторов
- 5) Переменные в Python, их особенности.
- 6) Типы данных Python
- 7) Запись математических операций. Особенности.
- 8) Инструкция import, from.
- 9) Операторы присваивания, виды, особенности, обмен ссылок
- 10) Функция для ввода и вывода данных. Форматный вывод.
- 11) Расширение файла программы на языке Python.
- 12) Запись операторов сравнения. Оператор is.
- 13) Числовые эквиваленты значений True и False.
- 14) Функция bool().
- 15) Логические операторы.
- 16) Приоритет операций и операторов.
- 17) Условный оператор языка Python. Виды.
- 18) Инструкция pass в условном операторе.

Образец заданий для выполнения лабораторных работ и примерный перечень вопросов для их защиты
«Решение задач циклической структуры»

Задание: Составить программу по варианту для решения задач

<p>Вычислить значение функции на отрезке с помощью операторов цикла while и for.</p> $f = ax^3 - bx \quad a = -2 \quad b = 1.7 \quad x \in [-2; 4] \quad h = 0.5$
<p>В трёхзначном числе зачеркнули первую цифру слева; когда полученное двузначное число умножили на 7, получилось исходное трёхзначное число. Найдите это число. Выведите это число в шестнадцатеричном и восьмеричном представлении. Составить блок-схему к программе.</p>
<p>Вычислить сумму n слагаемых</p> $\sin x + \sin^2 x + \sin^3 x + \sin^4 x + \dots + \sin^n x$
<p>Богатый дядюшка подарил мне 1 доллар в мой первый день рождения. В каждый следующий день рождения он удваивал сумму подарка и прибавлял к ней столько долларов, сколько лет мне исполнилось. Написать программу, подсчитывающую общую сумму денег, подаренных к N-му дню рождения.</p>

Контрольные вопросы:

- 1) Оператор цикла while. Общий вид, особенности работы. Блок-схема.
- 2) Оператор цикла for. Общий вид, особенности работы. Блок-схема.
- 3) Функция range.
- 4) Оператор break, continue. Применение.
- 5) Особенности использования чисел в логическом выражении оператора while.
- 6) Логические значения объектов.
- 7) Функции ceil, floor, trunc.
- 8) Особенности решения задач методом перебора. Примеры задач.

- 9) Функции Fraction, Decimal.
- 10) Представление чисел в разных системах счисления.

Образец заданий для выполнения лабораторных работ и примерный перечень вопросов для их защиты
«Решение задач с использованием коллекций Python»

Задание:

- 1) Создать список, элементы которого содержат только согласные буквы вашего ФИО. Для этого используйте генератор списков и список, состоящий из букв вашего ФИО, полученный с помощью функции list.
- 2) Создайте игру «Угадай пароль». На экране пользователь видит символы, из которых может состоять пароль (укажите не более 4 символов). Затем пользователь отвечает на вопрос: «Вы будете играть в «Угадай пароль?»». При положительном ответе программа случайным образом генерирует количество (kol) символов пароля (не больше 4 символов). На экране появляется сообщение, что пароль состоит из kol символов. Пользователь вводит пароль. При правильном пароле выходит сообщение «Вы угадали пароль!». Если пароль не верен, то у пользователя спрашивают о продолжении игры. Если пользователь выходит из игры, то выводится сообщение: «Вы не смогли угадать пароль!». Программу составляете с помощью списков. Игру можете модифицировать и сделать более интересной.
- 3) Создать словарь, используемый для перевода слов с русского на английский или наоборот. Не менее 10 слов. Реализовать возможность добавления новых слов в словарь, вывода списка слов в столбец, проверки наличия слова для перевода в словаре. Также можете предусмотреть другие возможности.
- 4) Создать словарь с помощью генератора словарей. При защите пояснить как формируется словарь.
- 5) Создать словарь с помощью вложенных генераторов. При защите пояснить как формируется словарь

Контрольные вопросы:

- 1) Что такое коллекции?
- 2) Способы создания строк, списков, кортежей
- 3) Функция range. Назначение
- 4) Общие методы работы с последовательностями.
- 5) Что такое отображения?
- 6) Кортеж в Python. Способы задания. Создания кортежа из одного элемента.
- 7) Определение позиции элемента в кортеже.
- 8) Привести примеры операций, функций с кортежами являющихся общими для всех последовательностей.
- 9) Словарь. Способы задания словаря.
- 10) Генераторы словарей.
- 11) Глубокое и поверхностное копирование в Python.

Образец заданий для выполнения лабораторных работ и примерный перечень вопросов для их защиты
«Решение задач на массивы (списки)»

Задание: Создать программы по вариантам

1. Создать массив, каждый элемент которого равен квадрату последующего индекса. Вывести полученный массив на экран. Вставьте в массив пятый элемент, значение которого равно 90. Вывести изменённый массив.

2. В массиве хранятся данные об осадках за месяц. Количество осадков получить с помощью функции *RND*. Диапазон значений выбираете самостоятельно.
 - a. Определить среднемесячное количество осадков.
 - b. Определить максимальное и минимальное количество осадков в году
 - c. Определить количество месяцев, в которых осадков было меньше чем в мае.
 - d. Сформировать массив и занести туда значения осадков в весенние и осенние месяцы.
3. Сформировать вектор, состоящий из 24 элементов. Имя вектора - первые буквы вашего имени.
 - Из вектора сформировать матрицу, состоящую из 4 строк и 6 столбцов. Имя матрицы - первые буквы вашей фамилии.
 - Определить минимальный элемент в каждом столбце матрицы.
 - Поменять местами 3 и 5 столбец матрицы
 - Вставить строку между 3 и 4 столбцом. Элементы строки – произведение индексов элементов. Вывести изменённую матрицу на экран.

Контрольные вопросы:

- 1) Что такое массив?
- 2) Как ввести элементы массива по формуле, с клавиатуры с помощью генератора псевдослучайных чисел?
- 3) Алгоритм сортировки массива
- 4) Алгоритм поиска максимального/минимального элемента в массиве
- 5) Алгоритм вставки строки, столбца в массив
- 6) Алгоритм удаления строки, столбца в массиве.

Образец заданий для выполнения лабораторных работ и примерный перечень вопросов для их защиты

«Функции пользователя в Python. Рекурсия.»

Задание: Написать программу для решения задачи с помощью функций пользователя
Сформировать вектор, состоящий из 30 элементов. Имя вектора - первые буквы вашего имени.

- Из вектора сформировать матрицу, состоящую из 5 строк и 6 столбцов. Имя матрицы - первые буквы вашей фамилии.
- Определить максимальный элемент в каждом столбце матрицы.
- Поменять местами 3 и 5 строку матрицы. Вывести изменённую матрицу на экран.
- Вставить строку между 3 и 4 строками. Элементы строки – сумма индексов элементов.
- Вывести изменённую матрицу на экран.

Образец заданий для выполнения лабораторных работ и примерный перечень вопросов для их защиты

«Работа с файлами и каталогами в Python»

Задание:

В пустом каталоге создать 9 текстовых файлов. Имена файлов пользователь вводит с клавиатуры. В каждый файл генерируется случайным образом последовательность из 3 однозначных чисел. Обратите внимание, что данные в файл записываются в виде строк (преобразуйте числа в строку). Вывести содержимое каталога на экран. Вывести содержимое одного из созданных файлов, который укажет пользователь. В один из созданных файлов в конец дописать свою Фамилию, Имя и Отчество.

Контрольные вопросы:

1. Что такое файл

2. С какими файлами может осуществляться работы в Python?
3. Перечислите этапы работы с файлами.
4. Функция открытия файла.
5. Какой метод используется для закрытия файла?
6. Какие существуют режимы доступа к файлу?
7. Приведите примеры использования разных режимов доступа к файлу.
8. Как правильно задать относительный и абсолютный путь доступа к файлу?
9. Перечислите методы работы с файлами
10. Как получить список файлов и каталогов в папке?
11. Чтение данных из файла с неизвестным количеством данных?
12. Приведите пример кода, когда метод close для закрытия файла не используется

Образец заданий для выполнения лабораторных работ и примерный перечень вопросов для их защиты
«Ошибки и исключения в Python»

Задание:

- 1) В пустом каталоге создать 6 текстовых файлов. Имена файлов пользователь вводит с клавиатуры. В каждый файл генерируется случайным образом последовательность из 5 однозначных чисел. Обратите внимание, что данные в файл записываются в виде строк (преобразуйте числа в строку). Вывести содержимое каталога на экран. Вывести содержимое одного из созданных файлов, который укажет пользователь. Запрос на указание имени файла производить до тех пор, пока пользователь не введёт правильное имя (работа с исключением).
- 2) Создать программу на обработку исключения при работе с функцией `sample`. Для пользователя указать описание и тип ошибки. Показать работу со всеми блоками при работе с исключением.

Контрольные вопросы:

1. Что такое исключение?
 2. Назовите типы ошибок в программе. Приведите примеры.
 3. Назовите функции и методы, при работе с которыми возбуждаются исключения.
 4. Общий вид инструкции для обработки исключения. Назначение блока `else` и `finally`.
- Примеры
5. Перечислите некоторые классы встроенных исключений.
 6. Как работают вложенные обработчики исключений. Примеры.
 7. Как получить информацию о возбуждаемом исключении в процессе выполнения программы?
 8. Способы закрытия файла.
 9. Менеджер контекста.

3.2 Типовые контрольные задания для проведения тестирования

Фонд тестовых заданий по дисциплине содержит тестовые задания, распределенные по разделам и темам, с указанием их количества и типа.

Структура фонда тестовых заданий по дисциплине

Индикатор достижения компетенции	Тема в соответствии с РПД	Характеристика ТЗ	Количество тестовых заданий, типы ТЗ
ОПК-2.2	Современные информационные технологии и программные средства	Знание	1-ОТЗ 1-ЗТЗ
		Умение	1-ОТЗ 1-ЗТЗ

		Действие	1-ОТЗ 1-ЗТЗ
ОПК-2.2 ОПК-7.1 ОПК-7.2 ОПК-7.3	Алгоритмы. Свойства алгоритмов. Блок-схемы	Знание	1-ОТЗ 1-ЗТЗ
		Умение	1-ОТЗ 1-ЗТЗ
		Действие	1-ОТЗ 1-ЗТЗ
ОПК-2.2 ОПК-7.1 ОПК-7.2 ОПК-7.3	Программные средства для создания алгоритмов в виде блок-схем (MS Visio, онлайн-инструменты)	Знание	1-ОТЗ 1-ЗТЗ
		Умение	1-ОТЗ 1-ЗТЗ
		Действие	1-ОТЗ 1-ЗТЗ
ОПК-2.2 ОПК-7.1 ОПК-7.2 ОПК-7.3	Парадигмы программирования. Языки программирования. Интегрированные среды разработки (IDE)	Знание	2-ОТЗ 2-ЗТЗ
		Умение	1-ОТЗ 1-ЗТЗ
		Действие	1-ОТЗ 1-ЗТЗ
ОПК-2.2 ОПК-7.1 ОПК-7.2 ОПК-7.3	Введение в язык программирования Python	Знание	2-ОТЗ 2-ЗТЗ
		Умение	2-ОТЗ 2-ЗТЗ
		Действие	2-ОТЗ 2-ЗТЗ
ОПК-2.2 ОПК-7.1 ОПК-7.2 ОПК-7.3	Условный оператор языка Python. Операторы сравнения. Логический тип данных.	Знание	2-ОТЗ 2-ЗТЗ
		Умение	2-ОТЗ 2-ЗТЗ
		Действие	2-ОТЗ 2-ЗТЗ
ОПК-2.2 ОПК-7.1 ОПК-7.2 ОПК-7.3	Операторы цикла в языке Python	Знание	2-ОТЗ 2-ЗТЗ
		Умение	2-ОТЗ 2-ЗТЗ
		Действие	2-ОТЗ 2-ЗТЗ
ОПК-2.2 ОПК-7.1 ОПК-7.2 ОПК-7.3	Коллекции Python. Последовательности. Виды копирования.	Знание	2-ОТЗ 2-ЗТЗ
		Умение	2-ОТЗ 2-ЗТЗ
		Действие	2-ОТЗ 2-ЗТЗ
ОПК-7.2 ОПК-7.3	Массивы (списки) на языке Python	Знание	2-ОТЗ 2-ЗТЗ
		Умение	2-ОТЗ 2-ЗТЗ
		Действие	2-ОТЗ 2-ЗТЗ
ОПК-2.2 ОПК-7.1 ОПК-7.2 ОПК-7.3	Множества в языке Python	Знание	2-ОТЗ 2-ЗТЗ
		Умение	2-ОТЗ 2-ЗТЗ

		Действие	2-ОТЗ 2-ЗТЗ
ОПК-2.2 ОПК-7.1 ОПК-7.2 ОПК-7.3	Функции пользователя в Python. Рекурсия.	Знание	2-ОТЗ 2-ЗТЗ
		Умение	2-ОТЗ 2-ЗТЗ
		Действие	2-ОТЗ 2-ЗТЗ
ОПК-2.2 ОПК-7.1 ОПК-7.2 ОПК-7.3	Функции map, filter, zip, reduce	Знание	2-ОТЗ 2-ЗТЗ
		Умение	2-ОТЗ 2-ЗТЗ
		Действие	2-ОТЗ 2-ЗТЗ
ОПК-2.2 ОПК-7.1 ОПК-7.2 ОПК-7.3	Работа с файлами и каталогами в Python	Знание	2-ОТЗ 2-ЗТЗ
		Умение	2-ОТЗ 2-ЗТЗ
		Действие	2-ОТЗ 2-ЗТЗ
ОПК-2.2 ОПК-7.1 ОПК-7.2 ОПК-7.3	Модули и пакеты в Python	Знание	2-ОТЗ 2-ЗТЗ
		Умение	2-ОТЗ 2-ЗТЗ
		Действие	2-ОТЗ 2-ЗТЗ
ОПК-2.2 ОПК-7.1 ОПК-7.2 ОПК-7.3	Ошибки и исключения в Python	Знание	2-ОТЗ 2-ЗТЗ
		Умение	2-ОТЗ 2-ЗТЗ
		Действие	2-ОТЗ 2-ЗТЗ
ОПК-2.2 ОПК-7.1 ОПК-7.2 ОПК-7.3	Основы объектно-ориентированного программирования в Python	Знание	2-ОТЗ 2-ЗТЗ
		Умение	2-ОТЗ 2-ЗТЗ
		Действие	2-ОТЗ 2-ЗТЗ
ОПК-2.2 ОПК-7.1 ОПК-7.2 ОПК-7.3	Создание приложений с графическим интерфейсом	Знание	2-ОТЗ 2-ЗТЗ
		Умение	2-ОТЗ 2-ЗТЗ
		Действие	2-ОТЗ 2-ЗТЗ
		Итого	70-ЗТЗ 70-ЗТЗ

Полный комплект ФТЗ хранится в электронной информационно-образовательной среде ИрГУПС и обучающийся имеет возможность ознакомиться с демонстрационным вариантом ФТЗ.

Ниже приведен образец типового варианта итогового теста, предусмотренного рабочей программой дисциплины.

1. Какое значение будет выведено на экран в результате выполнения фрагмента программы?

```
x = y = z = 5
m, n = 2, 3
y *= n
print(y)
```

Ответ: 15

2. Чему равно значение выражения `-6 ** 2` в Python ?

Ответ: -36

3. Какое значение будет получено в результате выполнения кода?

```
1 s = 0
2 for x in range(2, 6):
3     s += x
4 print (s)
```

Ответ: 14

4. Что будет выведено на экран в результате выполнения кода. Если значений несколько, то указывайте их через знак нижнего подчёркивания.

Например, 7_8 или A_5_6

```
2 def my(k):
3     k = copy.deepcopy(k)
4     k["b"] = 5
5     return k["b"]
6 sl = {"a":3, "b":7, "c":9}
7 t = my(sl)
8 print(t, sl["b"])
```

Ответ: 5_7

5. Дан фрагмент кода. Какие значения будут выведены на экран в результате выполнения кода?

```
b = ["ч", "д", "щ", "ь", "э", "я", "ю"]
for i in range(6):
    if i % 2 == 0 and i != 0:
        print(b[i])
```

Ответ: Щ_Э

6. Что будет выведено на экран, в результате выполнения фрагмента программы. Если выводится несколько значений, то указывайте их без пробела.

```
a = 2; b = 6; c = 1
print("{2}{1}{0}".format(a,b,c))
```

Ответ: 162

7. Какая базовая алгоритмическая структура предусматривает многократное повторение однотипных действий?

Ответ: Цикл

8. Конечный набор символов, с помощью которого могут быть образованы слова и величины языка программирования

Ответ: Алфавит

9. Какой результат будет получен в результате выполнения кода. При указании ответа учитывайте синтаксис языка Python.

```
>>> a=4; b=5
>>> (a % 2 == 0) ^ (b % 2 == 0)
```

Ответ: True

10) Какие математические выражение на языке Python записаны верно?

(модуль **math** импортирован с помощью инструкции `from math import *`)

A
`cos(x) ** 2`

B
`sin(x ** 2)`

C
`cos ** 2 (x)`

D
`|x|`

11) Будут ли являться результаты выполнения кодов одинаковыми?

```
1 x = 10
2 while x:
3     x = x - 1
4     if x % 2 != 0:
5         continue
6     print(x, end = ' ')
```

```
1 x = 10
2 while x:
3     x = x - 1
4     if x % 2 == 0:
5         print(x, end = ' ')
6
```

A) Да B) Нет

12) Какой блок при работе с исключениями перехватывает исключения?

- 1) finally
- 2) try**
- 3) else
- 4) except
- 5) close

13) В языке Python пока ложно условие **L** выполняется тело цикла **B**

```
While L :
B
```

a) верно b) не верно

14) Перечислите виды ошибок в программах.

- A **логические**
- B статистические
- C **ошибки времени выполнения**
- B **синтаксические**
- E пунктуационные

15) Получить первую цифру четырёхзначного числа **t** позволяет операция

- a) **t // 1000**
- b) t div 1000
- c) t % 100
- d) t \ 1000

16) Какие из представленных объектов являются неизменяемыми?

A	B	C
"привет"	[2, 3, "12", [3, 4]]	25

17) Какая из представленных функций позволяет применять функцию к каждому элементу последовательности (или объекту, поддерживающему итерирование)?

- A **map**
- B filter
- C zip
- B reduce
- E obj_iter

18) Сопоставьте между собой фразы

- | | | |
|-----------------------|---|--|
| A Поверхностная копия | 1 | Создаёт новый составной объект и затем (по мере необходимости) вставляет в него ссылки на объекты, находящиеся в оригинале |
| B Глубокая копия | 2 | Создаёт новый составной объект и затем вставляет в него копии объектов, находящихся в оригинале. |
| | 3 | Создаёт новый объект с указанием уровней вложения. |

Ответ: A-1 B-2

19) Аббревиатура IDE обозначает :

- i. **Интегрированная среда разработки**
- ii. Интегрированная система кодирования программ
- iii. Интегрированные языки программирования

3.3 Типовое задание для выполнения курсовой работы

Типовые задания выложены в электронной информационно-образовательной среде ИрГУПС, доступной обучающемуся через его личный кабинет.

Ниже приведен образец задания для выполнения курсовой работы и примерный перечень вопросов для ее защиты.

Образец типового задания для выполнения курсовой работы

Курсовая работа по дисциплине «Языки программирования» состоит из двух заданий. Задание 1. 1. Рассмотреть два языка программирования, провести их сравнение между собой по разным параметрам.

- История появления ЯП
- Базовое назначение ЯП
- Используемые IDE
- Рассмотреть язык в разных классификациях языков программирования (например, типизация, вид трансляции и т.д.)
- Структура программы
- Пример реализации базовых алгоритмических конструкций
- Анализ языка с точки зрения защиты данных (защита кода, использование в защите данных)
- Таблица сравнения языков программирования
- Реализация одной задачи на рассматриваемых языках.

План исследования представлен примерный. План исследования можно расширить.

Задание 2.

Разработать два модуля с функциями. В каждом модуле не меньше 8 функций. В каждом модуле функции используются для определённой области.

Например, - функции, которые используются в комбинаторике; - функции, представляющие различные алгоритмы шифрования; - функции, реализующие работу функций Python.

Например, запрограммировать работу функции count, методов, которые заменяют буквы в тексте со строчных на прописные без использования встроенных функций и т.д. Каждый модуль и функции должны быть документированы согласно правилам языка Python.

Образец типовых вопросов для защиты курсовых работ

1. К какому типу относятся рассматриваемые языки программирования по уровню абстракции?
2. К какой парадигме программирования относятся языки программирования?
3. Какая типизация поддерживается в языках программирования?
4. На какой платформе можно использовать язык программирования?
5. Как объявляются переменные в рассматриваемых языках программирования?
6. Какие перспективы развития языка программирования?

3.4 Перечень теоретических вопросов к экзамену

(для оценки знаний)

1. Алгоритм. Свойства алгоритма. Блоки для создания блок-схем алгоритма.
2. Языки программирования. Классификация языков программирования.
3. Языки программирования по различным классификациям. Примеры.
4. Парадигмы программирования.
5. Трансляторы, интерпретаторы, компиляторы.
6. Интегрированные среды разработки (IDE).
7. Язык Python и его особенности. IDE для написания программ на Python.
8. Особенности синтаксиса языка Python.

9. Математические операции на языке Python (сложение, умножение, деление, возведение в степень, целочисленное деление, нахождение остатка от деления) и их особенности.

10. Типы данных в Python. Определение типа объекта в Python.

11. Переменные в Python и их особенности.

12. Создание комментариев в Python.

13. Идентификаторы. Соглашение о наименовании.

14. Способы задания начальных значений для переменных.

15. Функция *input*. Синтаксис, назначение. Особенности использования.

16. Оператор присвоения. Виды оператора присвоения.

17. Обмен значение переменных (ссылок).

18. Импортирование библиотек и модулей. Инструкция *import, from*. Особенности использования.

19. Функция *print*. Параметры *sep, end*. Метод *format*. Особенности вывода целых, вещественных чисел.

20. Сокращённая форма записи оператора присвоения.

21. Операторы сравнения языка Python.

22. Логические значения *True* и *False*, числовые эквиваленты.

23. Функция для определения логического значения объекта.

24. Какие объекты возвращают значение *True* и *False*.

25. Оператор *is*. Примеры использования.

26. Логические операторы Python. Примеры записи.

27. Приоритет логических операций и операторов.

28. Условный оператор Python. Виды, блок-схемы.

29. Оператор *pass*. Назначение.

30. Тип данных *float* и его особенности. Модуль *fractions, decimal*.

31. Оператор цикла *while* в Python. Общий вид, блок-схема.

32. Реализация бесконечно цикла с помощью оператора *while*.

33. Оператор *break*. Назначение. Примеры использования.

34. Оператор *continue*. Примеры использования.

35. Функция *range*. Назначение. Особенности использования.

36. Оператор цикла *for*. Общий вид, блок-схема.

37. Нахождение суммы/произведения с помощью операторов цикла *while* и *for*.

38. Функции модуля *math: ceil, floor, trunc*.

39. Вычисление значений функций на отрезке с помощью операторов цикла *while* и *for*.

40. Примеры задач, решаемых методом перебора.

41. Последовательности Python. Общие операции, функции и методы, применяемые ко всем последовательностям. Примеры.

42. Понятие среза.

43. Строки в Python. Индексация элементов строки. Базовые операции со строками.

44. Функции для работы с символами.

45. Методы работы со строками. Понятие методы. Привести примеры методов.

46. Списки в Python. Способы создания списков. Синтаксис генератора списков. Примеры использования генератора списков.

47. Генерация вложенных списков.

48. Обращение к элементу списка. Обращение к элементам вложенных списков.

49. Применение общих операций и методов для работы с последовательностями к спискам.
50. Индексация. Срезы.
51. Методы работы со списками (привести примеры нескольких методов)
52. Добавление элемента в список, удаление элемента из списка.
53. Кортеж. Способы создания кортежей.
54. Словари. Правила при использовании словарей.
55. Способы создания словаря.
56. Генераторы словарей. Вложенные генераторы словарей.
57. Изменяемые и неизменяемые объекты. Глубокое и поверхностное копирование.
58. Функции получения псевдослучайных чисел.
59. Применение списков для работы с массивами одномерными и двумерными.
60. Обращение к элементам двумерного массива.
61. Задание элементов одномерного и двумерного массива.
62. Способы ввода элементов массива.
63. Вывод элементов двумерного массива в виде матрицы (2 способа).
64. Создание копий строки/столбца в двумерном массиве.
65. Реализация алгоритмов работы с одномерным массивом на языке Python:
 - a. Ввод /вывод элементов
 - b. Обмен местами элементов
 - c. Сортировка элементов
 - d. Вставка/удаление элемента
 - e. Определение суммы/произведения.
 - f. Определение суммы/произведения/количества элементов, удовлетворяющих условию.
 - g. Формирование вектора из элементов, удовлетворяющих условию.
 - h. Поиск максимального/минимального элемента и его номера
66. Реализация алгоритмов работы с двумерным массивом на языке Python:
 - a. Поиск максимального/минимального элемента в двумерном массиве и его индексов
 - b. Определение суммы/произведения элементов массива.
 - c. Определение количества/произведения/суммы элементов двумерного массива, удовлетворяющих заданному условию.
 - d. Определение количества/суммы/произведения/максимального/минимального элемента в строках/столбцах двумерного массива.
 - e. Формирование одномерного массива из элементов двумерного массива.
 - f. Формирование двумерного массива из элементов вектора
 - g. Обмен местами строк/столбцов двумерного массива
 - h. Сортировка элементов по строкам/столбцам в двумерном массиве
 - i. Удаление/вставка строки/столбца в двумерном массиве.
67. Подпрограммы Python. Определение. Назначение.
68. Задание функции пользователя. Синтаксис описания функции. Оператора *pass*.
69. Функции без параметров и с параметрами. Формальные и фактические параметры.
70. Расположение описаний функций.
71. Использование необязательных параметров в функции. Сопоставление параметров по ключам.

72. Особенности передачи значений в функцию из списка, кортежа, словаря.
73. Особенности задания функции с переменным числом параметров.
74. Особенности передачи объектов неизменяемого и изменяемого типа в функцию.
75. Анонимные функции (лямбда-функции).
76. Модули и пакеты в Python.
77. Создание модулей пользователя в Python.
78. Создание документации по функциям пользователя.
79. Работа с файлами в Python. Понятие файла. Этапы работы с файлами.
80. Функция *open*. Метод *close*.
81. Особенности режимов доступа к файлу.
82. Методы работы с файлами: write, writelines, read, readline, readlines, seek.
83. Чтение файла с неизвестным количеством данных.
84. Ошибки и исключения в Python
85. Основы объектно-ориентированного программирования.
86. Создание приложения с графическим интерфейсом.

3.5 Перечень типовых простых практических заданий к экзамену (для оценки умений)

1. Создайте блок-схему решения задачи: Вычислить значение функции на отрезке:
 $f = ax^3 - bx \quad a = -2 \quad b = 1.7 \quad x \in [-2; 4] \quad h = 0.5$
2. Напишите программу для:
 - определения последней цифры трёхзначного числа;
 - определения второй цифры четырёхзначного числа;
 - определения среднего арифметического чисел, введённых с клавиатуры;
3. Запишите с помощью одного оператора присвоения, что три переменные x, y, z являются ссылками на объект целого типа 5.
4. x=5, y=6. Используя один оператор присвоения, обменяйте значения переменных x и y.
5. Напишите программу нахождения суммы первых 10 натуральных чисел.
6. Создайте словарь, в котором будет находиться информация о вашем возрасте, имени и фамилии.
7. Что означает запись **a[0][3]** для списка с именем **a**?
8. Запишите сокращённую форму для оператора присвоения.
9. Вычислите значение выражения $\sqrt[3]{\frac{x^5+7}{|-6| \cdot y}}$.

```

1 s = 0
2 i = 2
3 while i <= 5:
4     s = s + i
5     i += 1
6 print (s)
```

10. Дан фрагмент программы

- сколько раз будет выполняться тело цикла?
- какое значение будет выведено на экран?

11. Что будет являться результатом выполнения фрагмента программы

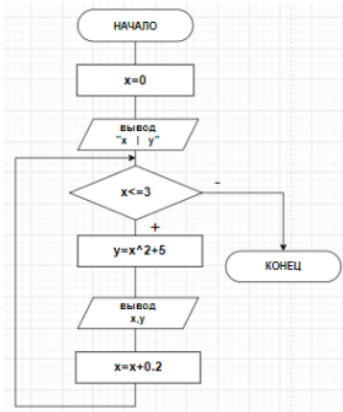
```

1 i = 100
2 while i:
3     print(i)
4     i -= 1
5

```

?

12. Создайте программу на языке Python по заданной блок-схеме.



3.6 Перечень типовых практических заданий к экзамену (для оценки навыков и (или) опыта деятельности)

1. Вычислить значение функции на отрезке с помощью операторов цикла **while** и **for**. Значение функции вывести в текстовый файл.

$$f = ax^3 - bx \quad a = -2 \quad b = 1.7 \quad x \in [-2; 4] \quad h = 0.5$$

2. В трёхзначном числе зачеркнули первую цифру слева; когда полученное двузначное число умножили на 7, получилось исходное трёхзначное число. Найдите это число. Выведите это число в шестнадцатеричном и восьмеричном представлении. Составить блок-схему к программе.

3. Вычислить сумму n слагаемых $\sin x + \sin^2 x + \sin^3 x + \sin^4 x + \dots + \sin^n x$.

4. Шахматная доска содержит 8×8 клеток. Горизонтали и вертикали обозначаются цифрами от 1 – 8. Написать программу, проверяющую, можно ли с первой заданной клетки попасть на вторую одним ходом ладьи.

5. С клавиатуры ввести путь доступа к файлу. В путь доступа к файлу входят не менее 5 папок. Вывести только имена папок на экран в виде списка.

6. Известен первый элемент массива A : $A[1]=6$. Создать массив, каждый элемент которого равен квадрату предыдущего индекса. Вывести полученный массив. Удалить третий элемент в массиве. Вывести изменённый массив.

10. В массиве хранятся данные, соответствующие росту учеников класса. Значение роста получить с помощью функции *RND*. Диапазон значений для роста укажите самостоятельно.

b. Определить средний рост учеников в классе.

- c. Определить рост самого низкого ученика в классе.
 - d. Определить сколько учеников отберут в баскетбольную секцию, если туда берут с ростом выше 170 см.
 - e. Вывести рост учеников класса в порядке возрастания.
11. Создать модуль пользователя с функциями для вычисления элементов треугольника.
 12. Используя генератор списков создать вектор, состоящий из чётных чисел, начиная с числа n . Сформировать матрицу из элементов вектора. Определить сумму элементов каждого столбца.
 13. Создать матрицу. Определить произведение элементов, значения которых меньше Z , в строках матрицы. Заменить все элементы в чётных строках на значение суммы всех элементов матрицы. Вывести значение суммы всех элементов в файл.
 14. Сформировать матрицу. Из элементов матрицы, значения которых больше заданного числа сформировать вектор. Найти произведение нечётных элементов столбцов матрицы.
 15. Привести пример программы с обработкой исключения.

4. Методические материалы, определяющие процедуру оценивания знаний, умений, навыков и (или) опыта деятельности

В таблице приведены описания процедур проведения контрольно-оценочных мероприятий и процедур оценивания результатов обучения с помощью оценочных средств в соответствии с рабочей программой дисциплины.

Наименование оценочного средства	Описания процедуры проведения контрольно-оценочного мероприятия и процедуры оценивания результатов обучения
Лабораторная работа	<p>Защита лабораторных работ проводится во время лабораторных занятий. Во время проведения защиты лабораторной работы пользоваться учебниками, справочниками, конспектами лекций, тетрадями не разрешено.</p> <p>Преподаватель на лабораторной работе, предшествующей занятию проведения защиты лабораторной работы, доводит до обучающихся: номер защищаемой лабораторной работы, время на защиту лабораторной работы.</p> <p>Преподаватель информирует обучающихся о результатах защиты лабораторной работы сразу после ее контрольно-оценочного мероприятия</p>
Курсовая работа	<p>Ход выполнения разделов курсовой работы в рамках текущего контроля оценивается преподавателем исходя из объемов выполненных работ в соответствие со шкалами оценивания. Преподаватель информирует обучающихся о результатах оценивания выполнения курсового проекта сразу после контрольно-оценочного мероприятия.</p> <p>В ходе защиты курсовой работы обучающийся делает доклад протяженностью 5 – 7 минут. Преподаватель ставит окончательную оценку за курсовую работу после завершения защиты, учитывая уровень ее защиты</p>

Для организации и проведения промежуточной аттестации составляются типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы.

Перечень теоретических вопросов и типовые практические задания разного уровня сложности для проведения промежуточной аттестации обучающиеся получают в начале семестра через электронную информационно-образовательную среду ИрГУПС (личный кабинет обучающегося).

Описание процедур проведения промежуточной аттестации в форме экзамена и оценивания результатов обучения

Промежуточная аттестация в форме экзамена проводится путем устного собеседования по билетам или в форме компьютерного тестирования.

При проведении промежуточной аттестации в форме собеседования билеты составляются таким образом, чтобы каждый из них включал в себя теоретические вопросы и практические задания.

Билет содержит: два теоретических вопроса для оценки знаний. Теоретические вопросы выбираются из перечня вопросов к экзамену; два практических задания: одно из них для оценки умений (выбирается из перечня типовых простых практических заданий к экзамену); другое практическое задание для оценки навыков и (или) опыта деятельности (выбираются из перечня типовых практических заданий к экзамену).


Распределение теоретических вопросов и практических заданий по экзаменационным билетам находится в закрытом для обучающихся доступе. Разработанный комплект билетов (25-30 билетов) не выставляется в электронную информационно-образовательную среду ИрГУПС, а хранится на кафедре-разработчике фондов оценочных средств.

На экзамене обучающийся берет билет, для подготовки ответа на экзаменационный билет обучающемуся отводится время в пределах 45 минут. В процессе ответа обучающегося на вопросы и задания билета, преподаватель может задавать дополнительные вопросы.

Каждый вопрос/задание билета оценивается по четырехбалльной системе, а далее вычисляется среднее арифметическое оценок, полученных за каждый вопрос/задание. Среднее арифметическое оценок округляется до целого по правилам округления

При проведении промежуточной аттестации в форме компьютерного тестирования вариант тестового задания формируется из фонда тестовых заданий по дисциплине случайным образом, но с условием: 50 % заданий должны быть заданиями открытого типа и 50 % заданий – закрытого типа.

Образец экзаменационного билета

 <p>ИрГУПС 20__-20__ учебный год</p>	<p>Экзаменационный билет № 1 по дисциплине «<u>Языки программирования</u>»</p>	<p>Утверждаю: Заведующий кафедрой « _____ » ИрГУПС _____</p>
<p>1. Трансляторы. Интерпретаторы, компиляторы. 2. Особенности передачи значений в функцию из списка, кортежа, словаря в Python. 3. Напишите программу для определения последней цифры трёхзначного числа 4. Создать матрицу. Определить произведение элементов, значения которых меньше Z, в строках матрицы. Заменить все элементы в чётных строках на значение суммы всех элементов матрицы. Вывести значение суммы всех элементов в файл.</p>		