

**КАЗАНСКИЙ КООПЕРАТИВНЫЙ ИНСТИТУТ (ФИЛИАЛ)
АВТОНОМНОЙ НЕКОММЕРЧЕСКОЙ ОБРАЗОВАТЕЛЬНОЙ
ОРГАНИЗАЦИИ ВЫСШЕГО ОБРАЗОВАНИЯ
ЦЕНТРОСОЮЗА РОССИЙСКОЙ ФЕДЕРАЦИИ
«РОССИЙСКИЙ УНИВЕРСИТЕТ КООПЕРАЦИИ»**

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

ИНТЕРПРЕТИРУЕМЫЕ ЯЗЫКИ ПРОГРАММИРОВАНИЯ

Специальность: 10.05.01 Компьютерная безопасность

Специализация: «Разработка защищенного программного обеспечения»

Формы обучения: очная

Объем дисциплины:

в зачетных единицах: 2 з.е.

в академических часах: 72 ак.ч.

Форма промежуточной аттестации: зачет

Рабочая программа по дисциплине «Интерпретируемые языки программирования» по специальности 10.05.01 Компьютерная безопасность, специализации «Разработка защищенного программного обеспечения», составлена в соответствии с требованиями Федерального государственного образовательного стандарта высшего образования – специалитет по специальности 10.05.01 Компьютерная безопасность, утвержденного приказом Министерства науки и высшего образования Российской Федерации от «26» ноября 2020 г. № 1459, профессионального стандарта 06.032 «Специалист по безопасности компьютерных систем и сетей», утвержденного приказом Министерства труда и социальной защиты Российской Федерации от 14.09.2022 г. № 533н, профессионального стандарта 06.033 «Специалист по защите информации в автоматизированных системах», утвержденного приказом Министерства труда и социальной защиты Российской Федерации от 14.09.2022 г. №525н.

Рабочая программа:

обсуждена и рекомендована к утверждению решением кафедры естественных дисциплин, сервиса и туризма от 25 марта 2025 г., протокол № 6

одобрена Научно-методическим советом Казанского кооперативного института (филиала) от 2 апреля 2025 г., протокол № 3

СОДЕРЖАНИЕ

1. Цель и задачи освоения дисциплины	4
2. Место дисциплины в структуре образовательной программы	5
3. Перечень планируемых результатов обучения по дисциплине	6
4. Объем дисциплины и виды учебной работы.....	8
5. Содержание дисциплины, структурированное по темам (разделам) с указанием количества академических часов и видов учебных занятий.....	8
5.1. Содержание дисциплины	8
5.2. Разделы, темы дисциплины и виды занятий	9
6. Практические занятия	10
7. Лабораторные работы	10
8. Примерная тематика курсовых работ (проектов)	11
9. Самостоятельная работа студента	11
10. Оценивание результатов обучения и уровня сформированности компетенций	13
11. Учебно-методическое обеспечение дисциплины	13
12. Перечень программного обеспечения, профессиональных баз данных и информационных справочных систем, необходимых для освоения дисциплины.....	15
13. Материально–техническое обеспечение дисциплины	16
Обновление рабочей программы дисциплины (модуля)	17

1. Цель и задачи освоения дисциплины

К группе интерпретируемых языков программирования относят языки, в которых операторы программы друг за другом отдельно транслируются и сразу выполняются (интерпретируются) с помощью специальной программы-интерпретатора (что противопоставляется компилируемому языку, в котором все операторы программы заранее оттранслированы в объектный код). Такой язык может реализовывать конструкции, позволяющие динамические изменения на этапе времени выполнения (модификация существовавших или создание новых подпрограмм). Эти конструкции затрудняют компиляцию и трансляцию на компилируемый язык.

Язык программирования Python – это универсальный язык программирования с динамической типизацией, который позволяет разрабатывать программы в соответствии с разными парадигмами: процедурным программированием, объектно-ориентированным, параметрическим, функциональным и метапрограммированием. В данном курсе рассматриваются все основные возможности языка Python и их применение при разработке программ. Дается описание библиотек языка Python, необходимых для создания широкого круга программ.

Воспитательная цель: формирование свободного и творческого подхода к программированию на современных языках высокого уровня, интереса к наблюдению за тенденциями и новостями в области средств разработки программного обеспечения.

Отбор материала основывается на необходимости ознакомить студентов со следующей современной научной информацией:

- об особенностях и последних достижениях в области разработки кроссплатформенного ПО;
- о различных подходах к программированию, реализованному в языке Python.

Содержательное наполнение дисциплины обусловлено общими задачами в подготовке бакалавра.

Научной основой для построения программы данной дисциплины является теоретико-прагматический подход в обучении.

Задачи дисциплины

- изучение языка программирования Python как мультипарадигменного языкового средства, достаточно полно отражающего современные концепции разработки ПО;
- формирование навыков создания приложений на языке Python;
- совершенствование и углубление навыков объектно-ориентированного и функционального программирования;
- знакомство с основными аспектами многопоточного программирования с использованием библиотек Python;
- знакомство с основами создания приложений для взаимодействия с базами данных на основе технологии DB API 2.0;
- изучение принципов написания веб-приложений на Python;

библиотек, реализующие шаблоны веб-страниц;
- изучение инструментов обработки данных с помощью языка Python

2. Место дисциплины в структуре образовательной программы

Дисциплина «Интерпретируемые языки программирования» относится к части, формируемой участниками образовательных отношений Блока 1 «Дисциплины (модули)» основной образовательной программы – программы специалитета по специальности 10.05.01 Компьютерная безопасность специализация «Разработка защищенного программного обеспечения».

Код компетенции	Предшествующие дисциплины (модули), практики	Изучаемые в текущем семестре дисциплины (модули), практики	Последующие дисциплины (модули), практики
ОПК-7	Объектно-ориентированное программирование Дискретное программирование Основы программирования Методы программирования	Интерпретируемые языки программирования	

3. Перечень планируемых результатов обучения по дисциплине

Изучение дисциплины направлено на формирование у обучающихся компетенций.

Формируемые компетенции (код и наименование компетенции)	Индикаторы достижения компетенций	Планируемые результаты обучения
ОПК-7 Способен создавать программы на языках высокого и низкого уровня, применять методы и инструментальные средства программирования для решения профессиональных задач, осуществлять обоснованный выбор инструментария программирования и способов организации программ	ОПК-7.1 Демонстрирует умение создавать программы на языках высокого уровня, применять методы и инструментальные средства структурного и объектно-ориентированного программирования для решения профессиональных задач	Знать Предметная область прикладной математики и информатики Отечественный и международный опыт решения актуальных и значимых задач прикладной математики и информатики Методы проведения экспериментов и наблюдений, обобщения и обработки информации при решении задач в области прикладной математики и информатики Предметная область и методы математического моделирования в естественных науках Отечественный и международный опыт в исследовании математических моделей в естественных науках Уметь Использовать существующие типовые решения и шаблоны проектирования программного обеспечения, применять знания, полученные в области математических и (или) естественных наук, и использовать их в профессиональной деятельности Разработка на основе знаний, полученных в области математических и (или) естественных наук, изменение и согласование архитектуры программного обеспечения Применять методы проведения экспериментов, использовать и адаптировать существующие математические методы и системы программирования для разработки и реализации алгоритмов решения прикладных задач Владеть Проведение наблюдений и измерений, составление их описаний и формулировка выводов при анализе решений задач прикладной математики и информатики Деятельность, направленная на решение задач аналитического характера, предполагающих выбор и многообразие актуальных способов решения задач, с использованием знаний, полученных в области математических и (или) естественных наук

Формируемые компетенции (код и наименование компетенции)	Индикаторы достижения компетенций	Планируемые результаты обучения
	ОПК-7.4 Демонстрирует умение создавать программы на языках высокого уровня, применять методы визуального объектного моделирования в области разработки программ и соответствующие инструментальные средства в процессе решения профессиональных задач	Знать: Цели и задачи проводимых исследований и разработок, значимые задачи прикладной математики и информатики Цели и задачи проводимых исследований и разработок, методы адаптации существующих математических методов и систем программирования для разработки и реализации алгоритмов решения прикладных задач Уметь: Анализировать входные данные при решении задач в области прикладной математики и информатики Использовать существующие типовые решения и шаблоны при анализе входных данных Применять методы и средства проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов, использовать и адаптировать существующие математические методы и системы программирования для разработки и реализации алгоритмов решения прикладных задач Применять методы проведения экспериментов, математическое моделирование для решения задач в области профессиональной деятельности Ориентироваться в современных алгоритмах компьютерной математики, применять методы и средства проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов Владеть: Деятельность, направленная на решение задач актуальные и значимые задачи прикладной математики и информатики аналитического характера, предполагающих выбор и многообразие актуальных способов решения задач Проектирование структур данных с использованием методов математического моделирования для решения задач в области профессиональной деятельности Разработка математически сложных алгоритмов, изменение и согласование архитектуры программного обеспечения с системным аналитиком и архитектором программного обеспечения

4. Объем дисциплины и виды учебной работы

Объем дисциплины и виды учебной работы в академических часах с выделением объема контактной работы обучающихся с преподавателем и самостоятельной работы обучающихся

очная форма обучения

Вид учебной деятельности	ак. часов	
	Всего	По семестрам
		6
1. Контактная работа обучающихся с преподавателем:	51	51
Аудиторные занятия, часов всего, в том числе:	50	50
• занятия лекционного типа	16	16
• занятия семинарского типа:	34	34
практические занятия	34	34
лабораторные занятия	-	-
в том числе занятия в форме практической подготовки	-	-
Консультации	0,5	0,5
Контактные часы на аттестацию в период экзаменационных сессий	0,5	0,5
2. Самостоятельная работа студентов всего, в том числе	21	21
- подготовка курсовой работы	-	-
- проработка учебного (теоретического) материала	10	10
- выполнение домашних заданий по практическим занятиям	10	10
- подготовка к зачету	-	-
Промежуточная аттестация: зачет	-	-
ИТОГО:	часов	72
общая трудоемкость	зач. ед.	2

5. Содержание дисциплины, структурированное по темам (разделам) с указанием количества академических часов и видов учебных занятий

5.1. Содержание дисциплины

Тема 1. Интерпретируемые языки программирования и их особенности

История Python. Типы данных. Синтаксис, операторы и управляющие конструкции. IPython и Jupyter-блокноты. PyCharm и Google Collab.

Тема 2. Функции, модули и пакеты

Области видимости. Аргументы. Рекурсия. Аннотация функций. Лямбда-функции. Декораторы. Генераторы. Создание, использование модулей и пакетов.

Тема 3. Классы и объектноориентированное программирование
Определение класса. Методы класса. Приватные методы и переменные.

Статические методы. Наследование. Множественное наследование

Тема 4. Потоки, процессы, асинхронное программирование
Многопоточное программирование. Модуль threading
и класс Thread. Потоки Timer. Класс RLock. Синхронизация потоков.
Модуль multiprocessing и класс Process.

Тема 5. Сетевое программирование
Модуль socket. Клиент и сервер, работающие без установления
соединения. Клиент и сервер, работающие с установлением соединения.
Фреймворк socketserver.

Тема 6. Работа с базами данных
Создание базы данных SQLite. Транзакции. Уровни изоляции.

Тема 7. Фреймворк Django
Модели. Миграции. Запись данных. Шаблоны и статические данные.
Формы.

Тема 8. Хранение данных и их обработка.
NumPy: массивы и векторные вычисления. Pandas. Kaggle

5.2. Разделы, темы дисциплины и виды занятий

очная форма обучения

№ п/п	Наименование раздела, темы дисциплины	Виды занятий, включая самостоятельную работу студентов (в часах)			Индикаторы достижения компетенций
		занятия лекционного типа	занятия семинарского типа / из них в форме практической подготовки (лабораторные занятия)	самос- тоятельная работа	
1	Тема 1. Интерпретируемые языки программирования и их особенности	2	4	2	ОПК-7, ОПК- 7.1, ОПК-7.4
2	Тема 2. Функции, модули и пакеты	2	4	2	ОПК-7, ОПК- 7.1, ОПК-7.4
3	Тема 3. Классы и объектноориентированное программирование	2	4	2	ОПК-7, ОПК- 7.1, ОПК-7.4
4	Тема 4. Потоки, процессы, асинхронное программирование	2	4	2	ОПК-7, ОПК- 7.1, ОПК-7.4
5	Тема 5. Сетевое программирование	2	4	2	ОПК-7, ОПК- 7.1, ОПК-7.4
6	Тема 6. Работа с базами данных	2	4	3	ОПК-7, ОПК- 7.1, ОПК-7.4

№ п/п	Наименование раздела, темы дисциплины	Виды занятий, включая самостоятельную работу студентов (в часах)			Индикаторы достижения компетенций
		занятия лекционного типа	занятия семинарского типа / из них в форме практической подготовки (лабораторные занятия)	самос- тоятельная работа	
7	Тема 7. Фреймворк Django	2	4	4	ОПК-7, ОПК-7.1, ОПК-7.4
8	Тема 8. Хранение данных и их обработка	2	6	4	ОПК-7, ОПК-7.1, ОПК-7.4
	Всего	16	34	21	

6. Практические занятия

№ п/п	Наименование раздела, темы дисциплины	Содержание лабораторных работ	Объем (час.)
			Очная форма обучения
1.	Тема 1. Интерпретируемые языки программирования и их особенности	Знакомство с python. IPython и Jupyter-блокноты. PyCharm и Google Collab	4
2.	Тема 2. Интерпретируемые языки программирования и их особенности	Типы данных. Списки. Множества. Словари.	4
3.	Тема 3. Функции, модули и пакеты	Лямбда-функции. Генераторы. Итераторы	4
4.	Тема 4. Классы и объектноориентированное программирование	Определение класса. Методы класса. Приватные методы и переменные. Статические методы.	4
5.	Тема 5. Классы и объектноориентированное программирование	Наследование. Множественное наследование. Декораторы в классах	4
6.	Тема 6. Потoki, процессы, асинхронное программирование	Многопоточное программирование. Multiprocessing.	4
7.	Тема 7. Потoki, процессы, асинхронное программирование	Асинхронное программирование	4
8.	Тема 8. Сетевое программирование	CGI-сервер	6
	Всего		34

7. Лабораторные работы

Лабораторные работы не предусмотрены

8. Примерная тематика курсовых работ (проектов)

Курсовые работы (проекты) не предусмотрены.

9. Самостоятельная работа студента

Самостоятельная работа студента при изучении дисциплины «Интерпретируемые языки программирования» направлена на:

- освоение нормативных правовых актов и учебной литературы, необходимых для освоения дисциплины;
- самостоятельный поиск информации в Интернете и других источниках;
- выполнение домашних заданий по практическим занятиям;
- подготовку к зачету.

Краткие рекомендации по выполнению самостоятельной работы:

Успешное усвоение дисциплины предполагает большой, упорный, серьезный, систематический труд студентов. Важнейшая его составная часть – выполнение разных видов самостоятельной работы.

1. Составление тематического конспекта на основе изученной основной и дополнительной учебной литературы. В тематическом конспекте за основу берется содержание темы, вопросы для обсуждения.

Этапы работы.

1.1 Конспектирование делается только после того, как прочитан или усвоен материал для конспектирования.

1.2. Необходимо мысленно или письменно составить план конспекта. По этому плану и будет строиться конспект далее.

1.3. Составление самого конспекта. Можно сказать, что конспект – это расширенные тезисы, дополненные рассуждениями и доказательствами, содержащимися в материалах для конспекта, а также собственными мыслями и положениями составителя конспекта.

Писать конспект рекомендуется четко и разборчиво. В конспекте можно выделять места текста в зависимости от их значимости. Для этого применяются различного размера буквы, подчеркивания, замечания на полях.

Типовые задания для самостоятельной работы

1. Дана последовательность натуральных чисел, завершающаяся числом 0. Определите количество строгих локальных максимумов в этой последовательности. (Элемент последовательности называется локальным максимумом, если он строго больше предыдущего и последующего элемента последовательности. Первый и последний элемент последовательности не являются локальными максимумами.)

2. Определите наименьшее расстояние между двумя локальными максимумами последовательности натуральных чисел, завершающейся числом 0. Если в последовательности нет двух локальных максимумов, выведите число 0.

3. Даны два списка чисел, которые могут содержать до 10000 чисел каждый. Выведите все числа, которые входят как в первый, так и во второй список в порядке возрастания.

4. Дан файл, содержащий текст на русском языке и некоторые два слова. Определить, сколько раз они встречаются в тексте и сколько из них — непосредственно друг за другом.

5. Даны файл, содержащий текст на русском языке, и некоторые буквы. Найти слово, содержащее наибольшее количество указанных букв.

6. Даны файл, содержащий текст на русском языке, и некоторая буква. Подсчитать, сколько слов начинается с указанной буквы.

7. Создать класс с полями, указанными в индивидуальном задании. Реализовать в классе методы: конструктор по умолчанию; функции обработки данных (1 и 2) функцию формирования строки информации об объекте. Создать класс-потомок с полями, указанными в индивидуальном задании. Реализовать в классе-потомке методы: конструктор; функцию обработки данных; функцию формирования строки информации об объекте.

Класс 1: Дата (три числа): день, месяц, год

Определить, является ли год високосным (кратным 4)

Увеличить дату на 5 дней

Класс 2: Человек: ФИО, телефон, дата рождения,

Количество дней до дня очередного рождения

8. Создать класс с полями, указанными в индивидуальном задании. Реализовать в классе методы: конструктор по умолчанию; функции обработки данных (1 и 2) функцию формирования строки информации об объекте. Создать класс-потомок с полями, указанными в индивидуальном задании. Реализовать в классе-потомке методы: конструктор; функцию обработки данных; функцию формирования строки информации об объекте.

Класс 1: Дата (три числа): день, месяц, год

Увеличить год на 1

Уменьшить дату на 2 дня

Класс 2: Работник: ФИО, дата поступления на предприятие

Количество лет работы на предприятии

9. Реализуйте подсчет определенного интеграла на интервале $[a, b]$ для функции $f(x)$ методом левых и правых прямоугольников.

10. Реализуйте подсчет определенного интеграла на интервале $[a, b]$ для функции $f(x)$ методом центральных прямоугольников.

11. Создать функцию, которая проверяет пингуются ли заданные в файле IP-адреса. Проверка IP-адресов должна выполняться параллельно в разных потоках.

12. Создать функцию, которая отправляет одну и ту же команду show на разные устройства в параллельных потоках, а затем записывает вывод команд в файл. Вывод с устройств в файле может быть в любом порядке.

13. Создать БД по заданной тематике. БД должна содержать не менее трех таблиц. Должна присутствовать возможность экспорта/импорта из/в xml-файл, а также не менее трех статистических запросов.

а) «Направления музыки».

б) «Направления танца».

в) «Фото-видео техника».

14. Создать web-приложение по заданной тематике с помощью фреймворка Django. Приложение должно предоставлять разный доступ к данным в зависимости от роли пользователя (только просмотр, просмотр/редактирование данных)

а) «Направления музыки».

б) «Направления танца».

в) «Фото-видео техника».

15. Используя датасет (load_boston):

а) получить график того, как зависит цена от криминогенной обстановки,

б) построить модель,

в) обучить,

г) получить ошибку модели,

д) осуществить поиск по сетке гиперпараметров с помощью GridSearchCV с целью подбора параметров и уменьшения ошибки.

16. Используя датасет (load_boston):

а) получить график того, как зависит цена от удаленности к реке,

б) построить модель,

в) обучить,

г) получить ошибку модели,

д) осуществить поиск по сетке гиперпараметров с помощью GridSearchCV с

целью подбора параметров и уменьшения ошибки.

10. Оценивание результатов обучения и уровня сформированности компетенций

Оценивание результатов обучения по дисциплине и уровня сформированности компетенций (части компетенции) осуществляется в рамках текущего контроля успеваемости и промежуточной аттестации в соответствии оценочным материалам.

11. Учебно-методическое обеспечение дисциплины

Основная литература:

1. Чернышев, С. А. Основы программирования на python : учебное пособие для вузов / С. А. Чернышев. — Москва : Издательство Юрайт, 2022. — 286 с. — (Высшее образование). — ISBN 978-5-534-14350-8. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/477353> (дата

обращения: 18.06.2024).

2. Федоров, Д. Ю. Программирование на языке высокого уровня Python :

учебное пособие для среднего профессионального образования / Д. Ю. Федоров. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2022. — 210 с. — (Профессиональное образование). — ISBN 978-5-534-12829-1. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/487079> (дата обращения: 18.06.2024).

3. Гниденко, И. Г. Технология разработки программного обеспечения : учебное пособие для среднего профессионального образования / И. Г. Гниденко, Ф. Ф. Павлов, Д. Ю. Федоров. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2024. — 248 с.

— (Профессиональное образование). — ISBN 978-5-534-18131-9. — Текст : электронный //

Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/539215> (дата обращения: 18.06.2024).

Для освоения дисциплины инвалидами и лицами с ограниченными возможностями здоровья имеются издания в электронном виде в электронно-библиотечных системах

«Библиоклуб» и «Юрайт».

Дополнительная литература:

1. Сузи, Р. А. Язык программирования Python : учебное пособие : [16+] / Р. А. Сузи. — 2-е изд., испр. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ) : Бином. Лаборатория знаний, 2007. — 327 с. — (Основы информационных технологий). — Режим доступа: по подписке. — URL: <https://biblioclub.ru/index.php?page=book&id=233288> дата обращения: 18.06.2024). — ISBN 978-5-9556-0109-0. — Текст : электронный.

2. Шелудько, В. М. Язык программирования высокого уровня Python: функции, структуры данных, дополнительные модули : [16+] / В. М. Шелудько. — Ростов-на-Дону ; Таганрог : Южный федеральный университет, 2017. — 108 с. : ил. — Режим доступа: по подписке. — URL: <https://biblioclub.ru/index.php?page=book&id=500060> (дата обращения: 18.06.2024). — Библиогр. в кн. — ISBN 978-5-9275-2648-2. — Текст : электронный.

3. Программные системы статистического анализа: обнаружение закономерностей в данных с использованием системы R и языка Python : [16+] / В. М. Волкова, М. А. Семенова,

Е. С. Четвертакова, С. С. Вожов. — Новосибирск : Новосибирский государственный технический университет, 2017. — 74 с. : ил., табл. — Режим доступа: по подписке. — URL: <https://biblioclub.ru/index.php?page=book&id=576496> (дата обращения: 18.06.2024). — Библиогр.: с. 48. — ISBN 978-5-7782-3183-2. — Текст : электронный.

Электронные библиотечные системы (ЭБС) и электронные образовательные ресурсы

1. Электронно-библиотечная система ZNANIUM <https://www.znanium.com>
2. Универсальная справочно-информационная полнотекстовая база данных периодических изданий «East View» (ИВИС) <https://dlib.eastview.com/>
3. Электронно-библиотечная система «BOOK.ru» www.book.ru
4. Электронно-библиотечная система «IPRbooks» <http://www.iprbookshop.ru/>
5. ЭБС «Национальный цифровой ресурс «Руконт» <https://lib.rucont.ru/search>
6. Образовательная платформа ЮРАЙТ <http://www.urait.ru>

Электронный каталог Университета – Автоматизированная информационная библиотечная система (АИБС «МегаПро»)

12. Перечень программного обеспечения, профессиональных баз данных и информационных справочных систем, необходимых для освоения дисциплины

Лицензионное программное обеспечение, в том числе отечественного производства:

- DsktpEdu ALNG LicSARk OLV F1 1Y Acdmc Ent (MS Windows) (подписка на ПО).

Свободно распространяемое программное обеспечение, в том числе отечественного производства:

- 7-Zip (свободный файловый архиватор с высокой степенью сжатия данных) (отечественное ПО).

- FastStone Image Viewer (программа для просмотра изображений в Microsoft Windows (лицензия бесплатного программного обеспечения)).

- Foxit Reader (Бесплатное прикладное программное обеспечение для просмотра электронных документов в стандарте PDF (лицензия бесплатного программного обеспечения)).

- Indigo (система программ для создания и проведения компьютерного тестирования знаний).

- Google Chrome, ЯндексБраузер (браузер).

- Adobe Acrobat Reader DC (• ПО для просмотра, печати и комментирования документов в формате PDF)

- Visual Studio Code (интегрированная среда разработки программного обеспечения)

Профессиональные базы данных не используются.

Информационные справочные системы:

- СПС КонсультантПлюс. Компьютерная справочная правовая система, широко используется учеными, студентами и преподавателями (подписка на

ПО)

Каждый обучающийся в течение всего периода обучения обеспечивается индивидуальным неограниченным доступом к электронно-библиотечной

13. Материально–техническое обеспечение дисциплины

Образовательный процесс обеспечен учебными аудиториями для проведения учебных занятий, а именно для проведения занятий лекционного типа, практических занятий, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, помещениями для самостоятельной работы студентов.

Учебная аудитория для проведения занятий лекционного типа

Рабочее место преподавателя (стол, стул), рабочие места обучающихся, доска для написания мелом, кафедра.

Технические средства обучения: персональный компьютер-1, мультимедийное оборудование (проектор, проекционный экран) - 1

Учебная аудитория для проведения практических занятий, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации.

Рабочее место преподавателя (стол, стул), рабочие места обучающихся, доска для написания мелом, кафедра.

Технические средства обучения: персональный компьютер-1, мультимедийное оборудование (проектор, проекционный экран) – 1.

Помещение для самостоятельной работы обучающихся, оснащенное компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду института.

Рабочие места обучающихся.

Технические средства обучения: персональный компьютер - 13.

Учебные аудитории соответствуют действующим противопожарным правилам и нормам, укомплектованы учебной мебелью.

Обновление рабочей программы дисциплины (модуля)

Наименование раздела рабочей программы, в который внесены изменения

(измененное содержание раздела)

обсуждено и рекомендовано к утверждению решением кафедры
наименование кафедры

от ____ ____ 20__ г., протокол № ____

одобрено Научно-методическим советом ____ от ____ ____ 20__ г.,
протокол № ____