

Документ подписан простой электронной подписью  
Информация о владельце:  
ФИО: Ястребов Олег Александрович  
Должность: Ректор  
Дата подписания: 25.05.2026 09:26:17  
Уникальный программный ключ:  
ca953a0120d891083f939673078ef1a989dae18a

**Федеральное государственное автономное образовательное учреждение высшего образования  
«Российский университет дружбы народов имени Патриса Лумумбы»**

**Институт русского языка**

(наименование основного учебного подразделения (ОУП) – разработчика ОП ВО)

## **РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ**

### **ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ КИБЕРТЕХНОЛОГИЙ В ГУМАНИТАРНОЙ СФЕРЕ**

(наименование дисциплины/модуля)

**Рекомендована МССН для направления подготовки/специальности:**

### **45.03.04 ИНТЕЛЛЕКТУАЛЬНЫЕ СИСТЕМЫ В ГУМАНИТАРНОЙ СФЕРЕ**

(код и наименование направления подготовки/специальности)

**Освоение дисциплины ведется в рамках реализации основной профессиональной образовательной программы высшего образования (ОП ВО):**

### **КИБЕРТЕХНОЛОГИИ И АНАЛИЗ ДАННЫХ В ГУМАНИТАРНОЙ СФЕРЕ**

(наименование (профиль/специализация) ОП ВО)

## 1. ЦЕЛЬ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Дисциплина «Программное обеспечение кибертехнологий в гуманитарной сфере» входит в программу бакалавриата «Кибертехнологии и анализ данных в гуманитарной сфере» по направлению 45.03.04 «Интеллектуальные системы в гуманитарной сфере» и изучается в 3, 4, 5, 6, 7 семестрах 2, 3, 4 курсов. Дисциплину реализует Кафедра прикладной информатики и интеллектуальных систем в гуманитарной сфере. Дисциплина состоит из 6 разделов и 61 тема и направлена на изучение методов, инструментов и технологий программирования, применяемых для разработки программного обеспечения в гуманитарной области с учетом требований кибербезопасности и анализа данных.

Целью освоения дисциплины является обучение студентов разрабатывать программное обеспечение, специализированное на решении задач гуманитарного характера с применением кибертехнологий. Это включает в себя умение проектировать и реализовывать программы и приложения, которые могут эффективно обрабатывать данные, обеспечивать их безопасность и проводить анализ для выявления паттернов, трендов и важной информации в гуманитарной сфере. В результате освоения этой дисциплины студенты должны усвоить навыки, необходимые для создания программного обеспечения, способного эффективно поддерживать деятельность организаций и проектов в гуманитарной сфере, а также внедрять инновационные решения для решения социальных проблем и улучшения качества жизни.

## 2. ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Освоение дисциплины «Программное обеспечение кибертехнологий в гуманитарной сфере» направлено на формирование у обучающихся следующих компетенций (части компетенций):

Таблица 2.1. Перечень компетенций, формируемых у обучающихся при освоении дисциплины (результаты освоения дисциплины)

Шифр	Компетенция	Индикаторы достижения компетенции (в рамках данной дисциплины)
УК-1	Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	УК-1.1 Анализирует задачу, выделяя её базовые составляющие; УК-1.2 Определяет и ранжирует информацию, требуемую для решения поставленной задачи; УК-1.3 Осуществляет поиск информации для решения поставленной задачи по различным типам запросов; УК-1.4 Работает с текстами различных типов, отличает факты от мнений, интерпретаций, оценок и обосновывает свои выводы с применением философского понятийного аппарата; УК-1.5 Анализирует и контекстно обрабатывает информацию для решения поставленных задач; УК-1.6 Предлагает варианты решения задачи, анализирует возможные последствия их использования с формированием собственных мнений и суждений;
ОПК-1	Способен применять в профессиональной деятельности методы математического анализа, логики и моделирования, теоретического и экспериментального исследования в информатике, лингвистике и гуманитарных науках	ОПК-1.1 Использует методы математического анализа, логики и моделирования в профессиональной деятельности; ОПК-1.2 Применяет знания теоретического и экспериментального исследования в информатике, лингвистике и гуманитарных науках;
ОПК-3	Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения	ОПК-3.2 Разрабатывает прикладные компьютерные программы, пригодные для практического применения в гуманитарной сфере; ОПК-3.3 Использует современный инструментальный для разработки алгоритмов и компьютерных программ, пригодных для практического применения в гуманитарной сфере;

### 3. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОП ВО

Дисциплина «Программное обеспечение кибертехнологий в гуманитарной сфере» относится к обязательной части блока 1 «Дисциплины (модули)» образовательной программы высшего образования.

В рамках образовательной программы высшего образования обучающиеся также осваивают другие дисциплины и/или практики, способствующие достижению запланированных результатов освоения дисциплины «Программное обеспечение кибертехнологий в гуманитарной сфере».

Таблица 3.1. Перечень компонентов ОП ВО, способствующих достижению запланированных результатов освоения дисциплины

Шифр	Наименование компетенции	Предшествующие дисциплины/модули, практики*	Последующие дисциплины/модули, практики*
УК-1	Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	Ознакомительная практика; Математическое обеспечение кибертехнологий в гуманитарной сфере; Математическое обеспечение кибертехнологий в гуманитарной сфере (избранные разделы); Базовый курс литературоведения; История России; Философия; Второй иностранный язык (практический курс);	Преддипломная практика;
ОПК-1	Способен применять в профессиональной деятельности методы математического анализа, логики и моделирования, теоретического и экспериментального исследования в информатике, лингвистике и гуманитарных науках	Ознакомительная практика; Математическое обеспечение кибертехнологий в гуманитарной сфере; Математическое обеспечение кибертехнологий в гуманитарной сфере (избранные разделы); Теория перевода;	Преддипломная практика;
ОПК-3	Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения	Ознакомительная практика; Введение в специальность. Аналитико-алгоритмическое обеспечение кибертехнологий;	Преддипломная практика;

\* - заполняется в соответствии с матрицей компетенций и СУП ОП ВО

\*\* - элективные дисциплины /практики

#### 4. ОБЪЕМ ДИСЦИПЛИНЫ И ВИДЫ УЧЕБНОЙ РАБОТЫ

Общая трудоемкость дисциплины «Программное обеспечение кибертехнологий в гуманитарной сфере» составляет «25» зачетных единиц.

Таблица 4.1. Виды учебной работы по периодам освоения образовательной программы высшего образования для очной формы обучения.

Вид учебной работы	ВСЕГО, ак.ч.	Семестр(-ы)	Семестр(-ы)	Семестр(-ы)	Семестр(-ы)	Семестр(-ы)
		3	4	5	6	7
Контактная работа, ак.ч	425	68	102	85	80	90
Лекции (ЛК)	179	34	34	34	32	45
Лабораторные работы (ЛР)	246	34	68	51	48	45
Практические/семинарские занятия (СЗ)	0	0	0	0	0	0
Самостоятельная работа обучающихся, ак.ч.	340	85	51	68	82	54
Контроль (экзамен/зачет с оценкой), ак.ч.	135	27	27	27	18	36
Общая трудоемкость дисциплины ак.ч.	ак.ч.	900	180	180	180	180
	зач.ед.	25	5	5	5	5

Общая трудоемкость дисциплины «Программное обеспечение кибертехнологий в гуманитарной сфере» составляет «25» зачетных единиц.

Таблица 4.2. Виды учебной работы по периодам освоения образовательной программы высшего образования для очно-заочной формы обучения.

Вид учебной работы	ВСЕГО, ак.ч.	Семестр(-ы)	Семестр(-ы)	Семестр(-ы)	Семестр(-ы)	Семестр(-ы)
		3	4	5	6	7
Контактная работа, ак.ч	260	34	51	51	64	60
Лекции (ЛК)	113	17	17	17	32	30
Лабораторные работы (ЛР)	147	17	34	34	32	30
Практические/семинарские занятия (СЗ)	0	0	0	0	0	0
Самостоятельная работа обучающихся, ак.ч.	523	119	111	93	116	84
Контроль (экзамен/зачет с оценкой), ак.ч.	117	27	18	36	0	36
Общая трудоемкость дисциплины ак.ч.	ак.ч.	900	180	180	180	180
	зач.ед.	25	5	5	5	5

## 5. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Таблица 5.1. Содержание дисциплины (модуля) по видам учебной работы\*

Номер раздела	Наименование раздела дисциплины	Наименование темы		Содержание темы	Вид учебной работы*
Раздел 1	NoSQL базы данных	1.1	Основные черты NoSQL баз данных.	Набор свойств BASE. Применение различных типов хранилищ. Возможность разработки базы данных без задание схемы. Линейная масштабируемость.	ЛК, ЛР
		1.2	Типы хранилищ данных.	Хранилище ключ-значение. Документно-ориентированная СУБД. Объектные хранилища данных.	ЛК, ЛР
		1.3	Особенности документо-ориентированных СУБД.	Документные хранилища. Документы и коллекции. Применение документно-ориентированных СУБД.	ЛК, ЛР
		1.4	Особенности и возможности MongoDB.	Установка и запуск MongoDB. JSON-формат. Поиск и обновление документов. Управление коллекциями и базами данных	ЛК, ЛР
		1.5	Моделирование данных в MongoDB.	Массивы и вложенные документы. Получение связанного документа. Денормализация	ЛК, ЛР
		1.6	Производительность и инструментарий MongoDB.	Шардинг. Репликация. Резервное копирование и восстановление	ЛК, ЛР
		1.7	Запросы в MongoDB, обработка документов.	Атомарная обработка документов. Агрегирование с помощью распределения-редукции и группировки. Обновление и удаление документов.	ЛК, ЛР
		1.8	Индексирование и оптимизация запросов.	Основы теории индексирования. Управление индексами. Оптимизация запросов.	ЛК, ЛР
Раздел 2	Разработка API	2.1	Введение в языковые модели.	Определение языковых моделей и их роль в обработке естественного языка (Natural Language Processing - NLP). История развития языковых моделей. Основные подходы к созданию языковых моделей	ЛК, ЛР
		2.2	Статистические методы в языковых моделях.	Основы статистических методов в обработке текста. Модели n-грамм: униграммы, биграммы, триграммы. Сглаживание и интерполяция в n-граммных моделях	ЛК, ЛР
		2.3	Методы машинного обучения для языковых моделей.	Обзор методов машинного обучения в контексте NLP. Рекуррентные нейронные сети (RNN) для последовательностей текста. Long Short-Term Memory (LSTM) и Gated Recurrent Unit (GRU) в языковых моделях	ЛК, ЛР
		2.4	Трансформеры и модели Attention.	Принцип работы трансформеров в обработке текста. Механизм внимания (attention mechanism) в языковых моделях. Применение трансформеров в моделях типа BERT, GPT, и других	ЛК, ЛР
		2.5	Генеративные модели текста.	Основы генеративных моделей в обработке текста. Автокодировщики (autoencoders) в языковых моделях. Генеративно-сопоставительные сети (GAN) для генерации текста	ЛК, ЛР
		2.6	Моделирование последовательностей.	Особенности моделирования последовательностей в языковых моделях. Seq2Seq модели: кодировщик-декодировщик (encoder-decoder) для задач машинного перевода. Attention в Seq2Seq моделях	ЛК, ЛР
		2.7	Применение языковых моделей в NLP задачах.	Анализ тональности текста. Извлечение информации из текста (NER - Named Entity Recognition). Машинный перевод и генерация текста	ЛК, ЛР
		2.8	Эволюция языковых моделей и их будущее.	Тенденции развития языковых моделей. Обзор актуальных исследований и применений языковых моделей. Этические и социальные вопросы использования языковых моделей	ЛК, ЛР
Раздел 3	Гибкие методологии	3.1	Введение в гибкие	Определение гибких методологий разработки ПО. История и основные принципы Agile	ЛК, ЛР

Номер раздела	Наименование раздела дисциплины	Наименование темы		Содержание темы	Вид учебной работы*
	разработки программного обеспечения		методологии.	и других гибких методологий. Сравнение гибких методологий с традиционными подходами к разработке	
		3.2	Scrum: основы и практики.	Основные концепции Scrum: итерации, роли (Product Owner, Scrum Master, Team). Практики Scrum: Sprint Planning, Daily Stand-ups, Sprint Review, Sprint Retrospective. Создание и управление бэклогом (Product Backlog)	ЛК, ЛР
		3.3	Kanban и Lean в гибкой разработке.	Принципы Kanban и Lean в разработке ПО. Канбан-доска и ее роль в визуализации процесса разработки. Continuous Delivery и Continuous Improvement в контексте Kanban	ЛК, ЛР
		3.4	Extreme Programming (XP) и технические практики.	Основные принципы XP: парное программирование, тестирование, рефакторинг. Тестирование в XP: Unit-тестирование, Test-Driven Development (TDD). Принципы кодирования и улучшения архитектуры	ЛК, ЛР
		3.5	Другие гибкие методологии: Feature-Driven Development (FDD), Crystal, Dynamic Systems Development Method (DSDM).	Основа каждой методологии и их применение в разработке ПО. Сравнение и преимущества различных подходов	ЛК, ЛР
		3.6	Масштабирование гибких методологий.	Scalable Agile Framework (SAFe) и его основные принципы. Large Scale Scrum (LeSS) и методы масштабирования Scrum для больших команд. Адаптация гибких методологий для работы в больших и распределенных командах	ЛК, ЛР
		3.7	Гибкие методологии вне разработки ПО.	Применение Agile в других областях: управление проектами, маркетинг, HR. Гибкие методологии в условиях переменчивости и неопределенности	ЛК, ЛР
		3.8	Эволюция и будущее гибких методологий.	Тенденции развития Agile: Agile at Scale, Agile в бизнесе. Влияние технологий (AI, IoT) на гибкие методологии. Этические и социальные вопросы в контексте гибких методологий	ЛК, ЛР
Раздел 4	Языки программирования и фреймворки	4.1	Основы языков программирования.	Введение в языки программирования: история, классификация. Основные концепции языков: синтаксис, семантика, типы данных. Применение различных языков для решения задач разного уровня сложности	ЛК, ЛР
		4.2	Структуры данных и алгоритмы.	Основные структуры данных: массивы, списки, стеки, очереди, деревья и графы. Алгоритмы сортировки, поиска, обхода и их реализация на выбранных языках	ЛК, ЛР
		4.3	Объектно-ориентированное программирование (ООП).	Принципы ООП: инкапсуляция, наследование, полиморфизм. Реализация ООП на примере выбранных языков: C++, Python	ЛК, ЛР
		4.4	Функциональное программирование.	Основы функционального программирования: функции высшего порядка, чистые функции. Применение функционального стиля в Haskell, JavaScript, Scala	ЛК, ЛР
		4.5	Языки разметки и запросов к данным.	Разработка веб-приложений с использованием HTML, CSS, JavaScript. Языки запросов к данным: SQL, NoSQL	ЛК, ЛР
		4.6	Введение в фреймворки и библиотеки.	Роль фреймворков и библиотек в разработке программного обеспечения. Примеры фреймворков: Django, React, Angular, Spring	ЛК, ЛР
		4.7	Многопоточное и асинхронное программирование.	Основы многопоточности и параллельного программирования. Асинхронное программирование: преимущества, примеры на Python, JavaScript	ЛК, ЛР
		4.8	Мобильная разработка и фреймворки.	Особенности мобильной разработки на iOS и Android. Фреймворки для мобильной разработки: Flutter, React Native, Xamarin	ЛК, ЛР
		4.9	Графические интерфейсы и	Разработка графических интерфейсов с использованием фреймворков: Qt, Tkinter,	ЛК, ЛР

Номер раздела	Наименование раздела дисциплины	Наименование темы	Содержание темы	Вид учебной работы*
		фреймворки.	Electron. Принципы и паттерны проектирования пользовательских интерфейсов	
		4.10 Тестирование и отладка программного обеспечения.	Основы тестирования: модульное, интеграционное, приемочное. Инструменты для тестирования и отладки кода на различных языках и платформах	ЛК, ЛР
Раздел 5	Проектирование реляционных баз данных	5.1 Основы реляционных баз данных.	Определение реляционных баз данных и их роль в хранении информации. История и эволюция реляционных моделей данных. Принципы работы реляционных баз данных и их отличия от других моделей	ЛК, ЛР
		5.2 Модель данных и структуры таблиц.	Концепции модели данных: сущности, атрибуты, отношения. Нормализация данных: первая, вторая, третья нормальные формы (1NF, 2NF, 3NF). Проектирование схемы таблиц и ключевых ограничений	ЛК, ЛР
		5.3 SQL и запросы к базам данных.	Основы SQL: SELECT, INSERT, UPDATE, DELETE. Сложные запросы: JOIN, подзапросы, агрегатные функции. Индексы и оптимизация запросов	ЛК, ЛР
		5.4 Нормализация и оптимизация баз данных.	Более глубокое понимание нормализации: BCNF, 4NF, 5NF. Денормализация данных: преимущества и недостатки. Оптимизация производительности баз данных: индексы, кэширование, партиционирование	ЛК, ЛР
		5.5 Транзакции и управление целостностью данных.	Основы транзакций: ACID-свойства (Atomicity, Consistency, Isolation, Durability). Управление параллельностью и блокировками. Роли и права доступа пользователей к данным	ЛК, ЛР
		5.6 Безопасность и защита данных.	Механизмы аутентификации и авторизации в реляционных базах данных. Защита от угроз: SQL-инъекции, атаки по переполнению буфера и другие. Роли SSL/TLS для обеспечения безопасности	ЛК, ЛР
		5.7 Бэкапы, восстановление и мониторинг.	Стратегии резервного копирования данных: полная, дифференциальная, инкрементальная. Практики восстановления данных и систем мониторинга баз данных. Планирование ежедневных операций обслуживания баз данных	ЛК, ЛР
		5.8 Эволюция баз данных и будущие тенденции.	Тенденции развития реляционных баз данных: NewSQL, распределенные базы данных. Влияние технологий (Big Data, IoT) на эволюцию реляционных баз данных. Этические и правовые аспекты использования реляционных баз данных	ЛК, ЛР
Раздел 6	Языки программирования	6.1 Алгоритмизация и программирование: основные понятия.	Алгоритм. Данные ЭВМ. Свойства алгоритма. Формы представления алгоритма. Процесс алгоритмизации. Алгоритмический язык и язык программирования. Программа. Понятие оператора.	ЛК, ЛР
		6.2 Система программирования.	Языки программирования. Классификация языков программирования. назначение, краткая характеристика. Система программирования: основные компоненты. Понятие машинного кода. Представление о трансляции и компиляции и интерпритации кода	ЛК, ЛР
		6.3 Данные и их типы: характеристика и классификация.	Характеристика данных в программировании. Классификация данных. Переменные и константы: различия в обработке при компиляции. Принцип обработки идентификатора транслятором. Типы данных. Назначение типа данных в программировании. Классификация типов данных.	ЛК, ЛР
		6.4 Типы вычислительных процессов.	Линейный вычислительный процесс. Типы вычислительных процессов: линейный ветвящийся и циклический. Управляющие кононические конструкции языка программирования. Принцип структурного программирования. Характеристика линейного вычислительного процесса. Организация ввода-вывода	ЛК, ЛР
		6.5 Выражение и операции.	Понятие выражения. Построение выражения и особенности его использования. Классификация и приоритет операций. Операция присваивания. Примеры использования основных операций.	ЛК, ЛР

Номер раздела	Наименование раздела дисциплины	Наименование темы		Содержание темы	Вид учебной работы*
		6.6	Ветвящийся процесс: ветвление.	Характеристика ветвящегося процесса. Полный и неполный варианты организации ветвления. Оператор и операция ветвления. Примеры использования. Вложенность оператора ветвления. Построение таблицы трассировки для вложенных ветвлений.	ЛК, ЛР
		6.7	Ветвящийся процесс: выбор из нескольких альтернатив.	Организация выбора решения из нескольких вариантов с использованием операторов if и switch. Особенности использования break внутри оператора switch/case.	ЛК, ЛР
		6.8	Циклический вычислительный процесс.	Цикл: основные понятия и каноническое представление. Классификация циклов. Цикл с пред и пост условиями. Особенности организации цикла с параметром. Сравнительный анализ работы операторов цикла. Примеры.	ЛК, ЛР
		6.9	Простые и вложенные циклы.	Принцип выполнения вложенных циклов. Альтернативная организация вычислительного процесса решения задачи на основе простых и вложенных циклов. Построение таблицы трассировки для вложенных циклов. Примеры задач, содержащие вложенные циклы: принцип попарного сравнения элементов одного множества.	ЛК, ЛР
		6.10	Числовые рекуррентные последовательности.	Способы определения числовой последовательности в программировании. Рекуррентное соотношение. Обобщенный член, начальное условие, порядок рекуррентного соотношения. Решение рекуррентного уравнения. Использование рекуррентных соотношений и обобщенного члена в программировании. Примеры программ с использованием рекуррентных соотношений. Нахождение значения с заданной точностью. Переход от обобщенного члена к рекуррентному соотношению при нахождении суммы ряда с заданной точностью. Сравнительный анализ их практического использования. Примеры программ.	ЛК, ЛР
		6.11	Функции: принцип декомпозиции задачи.	Принцип декомпозиции при решении задачи. Декомпозиция программы на подпрограммы (функции), пример. Область видимости переменных: глобальные и локальные переменные. Модуль. Особенности описания функций: заголовок и реализация. Фактические и формальные параметры. Пример реализации функции с параметрами. Реализация логических функций.	ЛК, ЛР
		6.12	Функции: особенности передачи параметров.	Способы передачи и получения значений при вызове функций. Особенности реализации.	ЛК, ЛР
		6.13	Массивы как параметры функций.	Массивы как параметры функций. Использование массивов в функциях.	ЛК, ЛР
		6.14	Приемы работы со строковым типом данных.	Символьные строки. Объявление строк в программе. Особенности организации в памяти. Функции работы со строками. Передача параметров. Копирование и конкатенация строк. Поиск подстроки. Примеры.	ЛК, ЛР
		6.15	Генератор псевдослучайных чисел.	Рекурсия. Организация псевдослучайной последовательности. Генератор псевдослучайной последовательности. Пример использования. Рекурсивный вызов функций. Дерево рекурсивных вызовов. Завершение рекурсивного вызова. Пример рекурсивных алгоритмов. Подведение итогов семестровой работы.	ЛК, ЛР
		6.16	Основы объектно-ориентированного программирования (ООП).	Принципы ООП: Инкапсуляция, наследование, полиморфизм. Классы и объекты в Python: Создание классов, экземпляры классов, атрибуты и методы.	ЛК, ЛР
		6.17	Наследование и полиморфизм в Python.	Наследование: Использование суперклассов и подклассов для организации кода. Полиморфизм: Работа с методами, имеющими одинаковые имена в разных классах.	ЛК, ЛР

Номер раздела	Наименование раздела дисциплины	Наименование темы		Содержание темы	Вид учебной работы*
		6.18	Инкапсуляция и абстракция.	Инкапсуляция в Python: Скрытие данных и методов, защита доступа к атрибутам. Абстракция и интерфейсы: Создание абстрактных классов, определение интерфейсов.	ЛК, ЛР
		6.19	Применение ООП в разработке.	Проектирование с использованием ООП: Моделирование объектов и их взаимодействие. Работа с библиотеками: Применение ООП в библиотеках и фреймворках Python.	ЛК, ЛР

\* - заполняется только по ОЧНОЙ форме обучения: ЛК – лекции; ЛР – лабораторные работы; СЗ – практические/семинарские занятия.

## 6. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Таблица 6.1. Материально-техническое обеспечение дисциплины

Тип аудитории	Оснащение аудитории	Специализированное учебное/лабораторное оборудование, ПО и материалы для освоения дисциплины (при необходимости)
Лекционная	Аудитория для проведения занятий лекционного типа, оснащенная комплектом специализированной мебели; доской (экраном) и техническими средствами мультимедиа презентаций.	
Компьютерный класс	Компьютерный класс для проведения занятий, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, оснащенная персональными компьютерами (в количестве 17 шт.), доской (экраном) и техническими средствами мультимедиа презентаций.	
Семинарская	Аудитория для проведения занятий семинарского типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, оснащенная комплектом специализированной мебели и техническими средствами мультимедиа презентаций.	
Для самостоятельной работы	Аудитория для самостоятельной работы обучающихся (может использоваться для проведения семинарских занятий и консультаций), оснащенная комплектом специализированной мебели и компьютерами с доступом в ЭИОС.	

\* - аудитория для самостоятельной работы обучающихся указывается **ОБЯЗАТЕЛЬНО!**

## 7. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

### Основная литература:

1. Чернышев, С. А. Принципы, паттерны и методологии разработки программного обеспечения : учебное пособие для вузов / С. А. Чернышев. — Москва : Издательство Юрайт, 2024. — 176 с. — (Высшее образование). — ISBN 978-5-534-14383-6.
2. Федоров, Д. Ю. Программирование на языке высокого уровня Python : учебное пособие для среднего профессионального образования / Д. Ю. Федоров. — 5-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2024. — 227 с. — (Профессиональное образование). — ISBN 978-5-534-17319-2.
3. Черткова, Е. А. Программная инженерия. Визуальное моделирование программных систем : учебник для среднего профессионального образования / Е. А. Черткова. — 3-е изд., испр. и доп. — Москва : Издательство Юрайт, 2024. — 146 с.
4. Лаврищева, Е. М. Программная инженерия и технологии программирования сложных систем : учебник для вузов / Е. М. Лаврищева. — 2-е изд., испр. и доп. — Москва : Издательство Юрайт, 2023. — 432 с. — (Высшее образование). — ISBN 978-5-534-07604-2.
5. 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.
6. Тузовский, А. Ф. Объектно-ориентированное программирование : учебное пособие для вузов / А. Ф. Тузовский. — Москва : Издательство Юрайт, 2024. — 213 с. — (Высшее образование). — ISBN 978-5-534-16316-2.

### Дополнительная литература:

1. Казарин, О. В. Надежность и безопасность программного обеспечения : учебное пособие для вузов / О. В. Казарин, И. Б. Шубинский. — Москва : Издательство Юрайт, 2024. — 342 с. — (Высшее образование). — ISBN 978-5-534-05142-1.

2. Новожилов, О. П. Информатика в 2 ч. Часть 1 : учебник для среднего профессионального образования / О. П. Новожилов. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2024. — 320 с. — (Профессиональное образование). — ISBN 978-5-534-06372-1.

*Ресурсы информационно-телекоммуникационной сети «Интернет»:*

1. ЭБС РУДН и сторонние ЭБС, к которым студенты университета имеют доступ на основании заключенных договоров

- Электронно-библиотечная система РУДН – ЭБС РУДН <https://mega.rudn.ru/MegaPro/Web>
- ЭБС «Университетская библиотека онлайн» <http://www.biblioclub.ru>
- ЭБС Юрайт <http://www.biblio-online.ru>
- ЭБС «Консультант студента» [www.studentlibrary.ru](http://www.studentlibrary.ru)
- ЭБС «Знаниум» <https://znanium.ru/>

2. Базы данных и поисковые системы

- Sage <https://journals.sagepub.com/>
- Springer Nature Link <https://link.springer.com/>
- Wiley Journal Database <https://onlinelibrary.wiley.com/>
- Наукометрическая база данных Lens.org <https://www.lens.org>

*Учебно-методические материалы для самостоятельной работы обучающихся при освоении дисциплины/модуля\*:*

1. Курс лекций по дисциплине «Программное обеспечение кибертехнологий в гуманитарной сфере».

\* - все учебно-методические материалы для самостоятельной работы обучающихся размещаются в соответствии с действующим порядком на странице дисциплины **в ТУИС!**

**РАЗРАБОТЧИКИ**

Заведующий кафедрой, к.т.н., доцент

---

Должность

**РУКОВОДИТЕЛЬ ОП ВО**

Заведующий кафедрой, к.т.н., доцент

---

Должность

**РУКОВОДИТЕЛЬ БУП**

Заведующий кафедрой, к.т.н., доцент

---

Должность

Софронова Е.А.

---

Фамилия И.О

Софронова Е.А.

---

Фамилия И.О

Софронова Е.А.

---

Фамилия И.О