

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Ястребов Олег Александрович
Должность: Ректор
Дата подписания: 02.06.2025 11:40:21
Уникальный программный ключ:
sa953a0120d891083f939673078ef1a989dae18a

Федеральное государственное автономное образовательное учреждение высшего образования
«Российский университет дружбы народов имени Патриса Лумумбы»
Факультет искусственного интеллекта
(наименование основного учебного подразделения (ОУП)-разработчика ОП ВО)

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

ИСТОРИЯ И ТЕОРИЯ ПРОГРАММИРОВАНИЯ

(наименование дисциплины/модуля)

Рекомендована МССН для направления подготовки/специальности:

**02.03.02 ФУНДАМЕНТАЛЬНАЯ ИНФОРМАТИКА И ИНФОРМАЦИОННЫЕ
ТЕХНОЛОГИИ,
09.03.03 ПРИКЛАДНАЯ ИНФОРМАТИКА**

(код и наименование направления подготовки/специальности)

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

**ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ: РАЗРАБОТКА И ОБУЧЕНИЕ
ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ**

(наименование (профиль/специализация) ОП ВО)

2025 г.

1. ЦЕЛЬ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Дисциплина «История и теория программирования» входит в программу бакалавриата «Искусственный интеллект: разработка и обучение интеллектуальных систем» по направлению 02.03.02 «Фундаментальная информатика и информационные технологии» и изучается в 1 семестре 1 курса. Дисциплину реализует Кафедра прикладного искусственного интеллекта. Дисциплина состоит из 3 разделов и 27 тем и направлена на изучение формирования у студентов целостного понимания эволюции программирования как научной и прикладной дисциплины, освоение фундаментальных принципов разработки программного обеспечения, стилей и парадигм программирования, а также на формирование теоретической базы для дальнейшего изучения современных языков и технологий программирования. Дисциплина служит вводным этапом, который подготавливает студентов к усвоению более специализированных дисциплин.

Целью освоения дисциплины является дать студентам представление об историческом развитии программирования, ключевых парадигмах, концепциях и теоретических основах программирования, сформировать понимание эволюции языков программирования и моделей вычислений, а также научить применять эти знания для анализа и сравнения существующих и разрабатываемых технологий программирования.

2. ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Освоение дисциплины «История и теория программирования» 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)
ОПК-2	Способен решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий, компьютерных/суперкомпьютерных методов и современного программного обеспечения, в том числе отечественного происхождения, с учетом основных требований информационной безопасности	ОПК-2.1 Обладает навыками разработки архитектуры программных систем и компонентов с учетом требований к производительности, надежности и безопасности; ОПК-2.2 Умеет применять информационно-коммуникационные технологии для поиска и анализа профессиональной информации.; ОПК-2.3 Знает основы информационной безопасности и методы защиты программного обеспечения от угроз и атак;
ОПК-6	Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, использовать их при решении задач профессиональной деятельности	ОПК-6.1 Знает базовые принципы цифровых технологий и методов, необходимых в профессиональной деятельности в области фундаментальной информатики и информационных технологий для: изучения и моделирования объектов профессиональной деятельности, анализа данных, представления информации и пр.; ОПК-6.2 Умеет применять необходимые в профессиональной деятельности цифровые технологии и методы в области фундаментальной информатики и информационных технологий для: изучения и моделирования объектов профессиональной деятельности, анализа данных, представления информации и пр.; ОПК-6.3 Владеет необходимыми в профессиональной деятельности технологиями и методами в области

Шифр	Компетенция	Индикаторы достижения компетенции (в рамках данной дисциплины)
		фундаментальной информатики и информационных технологий для: изучения и моделирования объектов профессиональной деятельности, анализа данных, представления информации и пр.;

3. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОП ВО

Дисциплина «История и теория программирования» 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 «История и теория программирования».

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

Шифр	Наименование компетенции	Предшествующие дисциплины/модули, практики*	Последующие дисциплины/модули, практики*
ОПК-6	Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, использовать их при решении задач профессиональной деятельности		Нейронные сети; Методы разработки решений на основе искусственного интеллекта (Git, Docker); Алгоритмы и структуры данных; Технологическая (проектно-технологическая) практика (учебная); Эксплуатационная практика (учебная); Технологическая (проектно-технологическая) практика (производственная); Эксплуатационная практика (производственная);
ОПК-2	Способен решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий, компьютерных/суперкомпьютерных методов и современного программного обеспечения, в том числе отечественного происхождения, с учетом основных требований информационной безопасности		Технологическая (проектно-технологическая) практика (производственная); Эксплуатационная практика (производственная); Технологическая (проектно-технологическая) практика (учебная); Эксплуатационная практика (учебная); Прикладные задачи

Шифр	Наименование компетенции	Предшествующие дисциплины/модули, практики*	Последующие дисциплины/модули, практики*
			машинного обучения; Программирование на языке Python; Методы машинного обучения; Основы глубокого обучения; Этика и безопасность использования искусственного интеллекта; Методы разработки решений на основе искусственного интеллекта (Git, Docker); Программирование на языке C++;

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

** - элективные дисциплины /практики

4. ОБЪЕМ ДИСЦИПЛИНЫ И ВИДЫ УЧЕБНОЙ РАБОТЫ

Общая трудоемкость дисциплины «История и теория программирования» составляет «3» зачетные единицы.

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)
			1
Контактная работа, ак.ч.	54		54
Лекции (ЛК)	18		18
Лабораторные работы (ЛР)	18		18
Практические/семинарские занятия (СЗ)	18		18
Самостоятельная работа обучающихся, ак.ч.	27		27
Контроль (экзамен/зачет с оценкой), ак.ч.	27		27
Общая трудоемкость дисциплины	ак.ч.	108	108
	зач.ед.	3	3

5. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

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

Номер раздела	Наименование раздела дисциплины	Содержание раздела (темы)		Вид учебной работы*
Раздел 1	История и эволюция программирования	1.1	Введение в историю программирования: появление понятий "программа" и "язык программирования"	ЛК
		1.2	Ранние языки программирования: ассемблеры, машинные коды, алгоритмические языки (Фортран, Лисп, Кобол)	ЛК
		1.3	Эволюция парадигм программирования: императивное, структурное, объектно-ориентированное, функциональное программирование	ЛК
		1.4	Анализ кода на ассемблере: структура и простейшие алгоритмы	ЛР
		1.5	Простейший проект на языке третьего поколения (Fortran/Pascal)	ЛР
		1.6	Сравнение стиля решения одной задачи на двух ранних языках программирования	ЛР
		1.7	Особенности первых программ и языков: факторы эволюции	СЗ
		1.8	История развития вычислительных машин и влияние на программирование	СЗ
		1.9	Дискуссия: почему возникали новые генерации языков программирования	СЗ
Раздел 2	Теоретические основы и парадигмы программирования	2.1	Формальные языки, грамматики и автоматы: основные понятия	ЛК
		2.2	Парадигмы программирования (императивная, объектно-ориентированная, функциональная, логическая)	ЛК
		2.3	Принципы структурного программирования и основные конструкции	ЛК
		2.4	Реализация алгоритмов на языке с поддержкой ООП: инкапсуляция, наследование	ЛР
		2.5	Пример функционального стиля: рекурсивное вычисление и композиция функций	ЛР
		2.6	Построение диаграммы грамматики небольшого языка программирования	ЛР
		2.7	Анализ алгоритмической сложности и базовых структур данных из теории программирования	СЗ
		2.8	Сравнение реализации одной задачи в разных парадигмах	СЗ
		2.9	Классификация языков программирования по парадигмам: группы, особенности, факты	СЗ
Раздел 3	Современные тенденции, анализ и перспективы разработки языков программирования	3.1	Трансляция, компиляция и интерпретация: традиционные и современные подходы	ЛК
		3.2	Безопасность и надежность языков программирования: типизация, управление памятью, автоматизация проверки	ЛК
		3.3	Текущие тренды и перспективы развития языков: новые парадигмы (реактивное, параллельное, визуальное программирование)	ЛК
		3.4	Компиляция и интерпретация: сравнение выполнения скрипта на интерпретируемом и компилируемом языке	ЛР
		3.5	Анализ системы типов на примере современных языков (например, Rust, Swift)	ЛР
		3.6	Мини-проект: анализ безопасности и	ЛР

Номер раздела	Наименование раздела дисциплины	Содержание раздела (темы)		Вид учебной работы*
			надежности кода на примере Open Source проекта	
		3.7	Обсуждение критериев выбора языка программирования для задачи искусственного интеллекта	СЗ
		3.8	Основные вызовы и сложности при создании новых языков и сред программирования	СЗ
		3.9	Прогноз развития языка программирования: взгляды экспертов, сравнение с тенденциями прошлого	СЗ

* - 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 25 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. Дильшоода Дилшод кизи Рустамова, Ринат Фаридович Бурнашев. Эволюция языков программирования / СамГИИЯ. // Science and Education. 2023. №4.5. С. 835-839. ISSN 2181-0842

Дополнительная литература:

1. Златопольский, Д. М. Программирование: типовые задачи, алгоритмы, методы: учебное пособие: [12+] / Д. М. Златопольский. – 4-е изд. (эл.). – Москва: Лаборатория знаний, 2020. – 226 с.: ил. – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=222873>. – Библиогр. в кн. – ISBN 978-5-00101-789-9. – Текст: электронный.

2. Грацианова, Т. Ю. Программирование в примерах и задачах: учебное пособие: [12+] / Т. Ю. Грацианова. – 6-е изд. – Москва: Лаборатория знаний, 2020. – 373 с.: ил., табл., граф. – (ВМК МГУ — школе). – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=448048>. – ISBN 978-5-00101-927-5. – Текст: электронный.

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

- 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/>

- Научометрическая база данных Lens.org <https://www.lens.org>

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

1. Курс лекций по дисциплине «История и теория программирования».

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

РАЗРАБОТЧИК:

Заведующий кафедрой
прикладного искусственного
интеллекта

Должность, БУП

Подпись

Подолько Павел
Михайлович

Фамилия И.О.

РУКОВОДИТЕЛЬ БУП:

Заведующий кафедрой
прикладного искусственного
интеллекта

Должность, БУП

Подпись

Подолько Павел
Михайлович

Фамилия И.О.

РУКОВОДИТЕЛЬ ОП ВО:

Заведующий кафедрой
прикладного искусственного
интеллекта

Должность, БУП

Подпись

Подолько Павел
Михайлович

Фамилия И.О.