

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение высшего
образования
«ИРКУТСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ»

Структурное подразделение «Институт информационных технологий и анализа данных
(121)»

УТВЕРЖДЕНА:

на заседании совета института ИТиАД им. Е.И.Попова
Протокол №8 от 16 февраля 2026 г.

Рабочая программа дисциплины

«ВВЕДЕНИЕ В ПРОГРАММИРОВАНИЕ»

Направление: 09.04.02 Информационные системы и технологии

Цифровизация промышленных предприятий

Квалификация: Магистр

Форма обучения: очная

Документ подписан простой
электронной подписью
Составитель программы:
Кононенко Роман
Владимирович
Дата подписания: 22.06.2026

Документ подписан простой
электронной подписью
Утвердил: Говорков Алексей
Сергеевич
Дата подписания: 22.06.2026

Документ подписан простой
электронной подписью
Согласовал: Кононенко Роман
Владимирович
Дата подписания: 22.06.2026

Год набора – 2026

Иркутск, 2026 г.

1 Перечень планируемых результатов обучения по дисциплине, соотнесённых с планируемыми результатами освоения образовательной программы

1.1 Дисциплина «Введение в программирование» обеспечивает формирование следующих компетенций с учётом индикаторов их достижения

Код, наименование компетенции	Код индикатора компетенции
ОПК-2 Способен разрабатывать оригинальные алгоритмы и программные средства, в том числе с использованием современных интеллектуальных технологий, для решения профессиональных задач	ОПК-2.2
ОПК-8 Способен осуществлять эффективное управление разработкой программных средств и проектов	ОПК-8.1

1.2 В результате освоения дисциплины у обучающихся должны быть сформированы

Код индикатора	Содержание индикатора	Результат обучения
ОПК-2.2	Понимает принципы работы современных информационных технологий и выполняет тестирование программных продуктов	Знать Основные принципы работы современных ИТ (архитектура ПО, сети, базы данных, облачные технологии) Этапы и виды тестирования программных продуктов (модульное, интеграционное, системное, приемочное) Инструменты для тестирования (unittest, pytest, PyTest, Selenium) Понятия: баг, дефект, ошибка, тест-кейс, тестовый сценарий Методы написания эффективных тестов Уметь Выполнять тестирование программных продуктов (модульное, функциональное, регрессионное) Писать простые тесты для проверки корректности кода Использовать инструменты автоматизированного тестирования (unittest, PyTest) Анализировать результаты тестирования и локализовать

		ошибки Составлять тестовые сценарии и тест-кейсы для проверки функциональности Владеть Навыками написания модульных тестов Инструментами отладки кода Способами автоматического и ручного тестирования Практикой тестирования программных продуктов на разных этапах разработки Методами составления отчетов по результатам тестирования
ОПК-8.1	Способен осуществлять тестирование разработанного программного обеспечения	Знать Уметь Владеть

2 Место дисциплины в структуре ООП

Изучение дисциплины «Введение в программирование» базируется на результатах освоения следующих дисциплин/практик: Нет

Дисциплина является предшествующей для дисциплин/практик: «Методы искусственного интеллекта в теории управления», «Научно-исследовательский семинар», «Прикладное применение машинного зрения»

3 Объем дисциплины

Объем дисциплины составляет – 4 ЗЕТ

Вид учебной работы	Трудоемкость в академических часах (Один академический час соответствует 45 минутам астрономического часа)	
	Всего	Семестр № 1
Общая трудоемкость дисциплины	144	144
Аудиторные занятия, в том числе:	28	28
лекции	14	14
лабораторные работы	0	0
практические/семинарские занятия	14	14
Самостоятельная работа (в т.ч. курсовое проектирование)	80	80
Трудоемкость промежуточной аттестации	36	36
Вид промежуточной аттестации (итогового контроля по дисциплине)	Экзамен	Экзамен

4 Структура и содержание дисциплины

4.1 Сводные данные по содержанию дисциплины

Семестр № 1

№ п/п	Наименование раздела и темы дисциплины	Виды контактной работы						СРС		Форма текущего контроля
		Лекции		ЛР		ПЗ(СЕМ)				
		№	Кол. Час.	№	Кол. Час.	№	Кол. Час.	№	Кол. Час.	
1	2	3	4	5	6	7	8	9	10	11
1	Основы программировани я и базовые конструкции	1	2			1	2			Тест
2	Функции и структуры данных	2	4			2	4			Тест
3	Работа с файлами и обработка данных	3	4			3	4	1	80	Тест
4	Основы тестирования программных продуктов	4	2			4	2			Тест
5	Современные информационные технологии и экосистема разработки	5	2			5	2			Тест
	Промежуточная аттестация								36	Экзамен
	Всего		14				14		116	

4.2 Краткое содержание разделов и тем занятий

Семестр № 1

№	Тема	Краткое содержание
1	Основы программирования и базовые конструкции	Введение в программирование: понятие алгоритма, переменные, типы данных, операторы. Условные конструкции (if-else) и циклы (for, while). Базовая структура программы. Понятие отладки и выполнения кода. Инструменты разработки (IDE, интерпретатор).
2	Функции и структуры данных	Определение и вызов функций, параметры и возвращаемые значения. Области видимости переменных. Базовые структуры данных: списки, словари, множества, кортежи. Методы работы со структурами. Итерация и базовые операции.
3	Работа с файлами и обработка данных	Чтение и запись файлов (текстовые, CSV, JSON). Обработка больших объемов данных. Исключения и обработка ошибок (try-except). Работа с датой и временем. Основы регулярных выражений для поиска и фильтрации данных.
4	Основы тестирования программных	Принципы и виды тестирования (модульное, интеграционное, системное). Написание тестов с

	продуктов	использованием unittest и pytest. Понятие багов, тест-кейсов и тестовых сценариев. Автоматизация тестирования. Отладка кода и анализ ошибок.
5	Современные информационные технологии и экосистема разработки	Принципы работы современных ИТ: клиент-серверная архитектура, базы данных, облачные технологии, контейнеризация (Docker). Системы контроля версий (Git). Инструменты совместной разработки. CI/CD: непрерывная интеграция и доставка. Тренды в разработке ПО.

4.3 Перечень лабораторных работ

Лабораторных работ не предусмотрено

4.4 Перечень практических занятий

Семестр № 1

№	Темы практических (семинарских) занятий	Кол-во академических часов
1	Разработка программы с использованием условных конструкций и циклов	2
2	Создание функций и работа со структурами данных	4
3	Обработка данных из файлов с использованием исключений	4
4	Написание модульных тестов для проверки корректности программы	2
5	Работа с Git и создание простого CI/CD пайплайна	2

4.5 Самостоятельная работа

Семестр № 1

№	Вид СРС	Кол-во академических часов
1	Подготовка к практическим занятиям	80

В ходе проведения занятий по дисциплине используются следующие интерактивные методы обучения: Интерактивное оборудование, интерактивная доска

5 Перечень учебно-методического обеспечения дисциплины

5.1 Методические указания для обучающихся по освоению дисциплины

5.1.1 Методические указания для обучающихся по практическим занятиям

Размещены на образовательной платформе el.istu.edu

5.1.2 Методические указания для обучающихся по самостоятельной работе:

Размещены на образовательной платформе el.istu.edu

6 Фонд оценочных средств для контроля текущей успеваемости и проведения промежуточной аттестации по дисциплине

6.1 Оценочные средства для проведения текущего контроля

6.1.1 семестр 1 | Тест

Описание процедуры.

Сдача практических работ, ответы на контрольные вопросы, прохождение теста

Критерии оценивания.

Отлично Самостоятельно разрабатывает тесты, выполняет полное тестирование, анализирует и исправляет ошибки, уверенно владеет инструментами автоматизации
Хорошо Выполняет тестирование с незначительной помощью, пишет базовые тесты, но испытывает трудности с анализом сложных ошибок
Удовлетворительно Выполняет тестирование по готовым сценариям, пишет простые тесты по шаблону, не способен разрабатывать сложные тест-кейсы
Неудовлетворительно Не понимает принципы тестирования, не может написать тесты, не способен анализировать ошибки в программных продуктах

6.2 Оценочные средства для проведения промежуточной аттестации

6.2.1 Критерии и средства (методы) оценивания индикаторов достижения компетенции в рамках промежуточной аттестации

Индикатор достижения компетенции	Критерии оценивания	Средства (методы) оценивания промежуточной аттестации
ОПК-2.2	Отлично Самостоятельно разрабатывает тесты, выполняет полное тестирование, анализирует и исправляет ошибки, уверенно владеет инструментами автоматизации Хорошо Выполняет тестирование с незначительной помощью, пишет базовые тесты, но испытывает трудности с анализом сложных ошибок Удовлетворительно Выполняет тестирование по готовым сценариям, пишет простые тесты по шаблону, не способен разрабатывать сложные тест-кейсы Неудовлетворительно Не понимает принципы тестирования, не может написать тесты, не способен анализировать ошибки в программных продуктах	Сдача практических работ
ОПК-8.1		

6.2.2 Типовые оценочные средства промежуточной аттестации

6.2.2.1 Семестр 1, Типовые оценочные средства для проведения экзамена по дисциплине

6.2.2.1.1 Описание процедуры

Ответы на контрольные вопросы из билета

Пример задания:

Что такое алгоритм? Какие свойства алгоритма вы знаете?
Какие типы данных существуют в языке программирования? Приведите примеры.
Что такое условный оператор? Как работает конструкция if-elif-else?
Какие виды циклов существуют? В чем разница между for и while?
Что такое функция? Как определить функцию и вызвать ее?
В чем разница между локальными и глобальными переменными?
Какие структуры данных вы знаете? Приведите примеры и опишите их назначение.
В чем отличие списка от словаря? Когда что удобнее использовать?
Что такое исключение? Как обрабатывать ошибки с помощью try-except?
Как выполняется чтение и запись данных из файла? Приведите пример.
Что такое тестирование программного обеспечения? Какие виды тестирования вы знаете?
Что такое модульное тестирование? Какие инструменты используются?
Что такое тест-кейс? Какие элементы он должен содержать?
Какие метрики используются для оценки качества тестирования?
Что такое Git? Какие основные команды Git вы знаете?
Что такое CI/CD? Как автоматизация тестирования встраивается в CI/CD-пайплайн?
Что такое клиент-серверная архитектура? Какие протоколы передачи данных используются?
В чем суть облачных технологий? Приведите примеры облачных сервисов.
Что такое контейнеризация? В чем преимущества Docker?
Опишите полный жизненный цикл разработки программного продукта: от идеи до поставки пользователю.

6.2.2.1.2 Критерии оценивания

Отлично	Хорошо	Удовлетворительно	Неудовлетворительно
Свободно владеет базовыми конструкциями языка (циклы, условия, функции, структуры данных) Самостоятельно разрабатывает программы, обрабатывает ошибки и работает	Знает конструкции языка, но иногда путается в сложных структурах Пишет программы по образцу, небольшие ошибки исправляет при подсказке Пишет базовые	Знает основные конструкции, но использует их шаблонно, без гибкости Пишет простые программы по готовым примерам, не всегда справляется самостоятельно Пишет простые	Не знает конструкции языка, не может написать простую программу Не понимает логику написания кода и тестов Не знаком с Git, CI/CD, облачными технологиями Путает базовые понятия (функция vs

с файлами Уверенно пишет модульные тесты (unittest/pytest) и анализирует результаты Понимает принципы работы ИТ (клиент-сервер, базы данных, облака, Git, CI/CD) Активно применяет Git, понимает CI/CD и процессы разработки Отвечает на все вопросы, решает задачи любой сложности	тесты, но сложные кейсы вызывают трудности Понимает принципы работы ИТ, но испытывает сложности с практическим применением Использует Git, но базово, без ветвления Отвечает на вопросы с незначительными уточнениями, большинство задач решает верно	тесты по шаблону, но не понимает логики проверок Поверхностно знает ИТ, не может объяснить принципы работы систем Работает с Git только по инструкции Отвечает на простые вопросы, сложные задачи решает с ошибками	процедура, список vs словарь, тест-кейс vs тест-план) Не может ответить на базовые вопросы, практические задания не выполняет или сдает с критическими ошибками
--	--	--	--

7 Основная учебная литература

1. Программирование на языке высокого уровня : методические указания по выполнению курсовой работы / Иркут. гос. техн. ун-т, 2007. - 27.

[Сайт] – URL: <http://elib.istu.edu/viewer/view.php?file=/files/er-3665.pdf>

2. Программирование на C++ : задания и методические указания по выполнению лабораторных работ / Иркут. гос. техн. ун-т, 2006. - 55.

[Сайт] – URL: <http://elib.istu.edu/viewer/view.php?file=/files/er-7426.pdf>

3. Немнюгин Сергей Андреевич. Turbo Pascal. Программирование на языке высокого уровня : [По направлению подгот. дипломир. специалистов "Информатика и вычисл. техника"] / С. А. Немнюгин, 2004. - 543.

8 Дополнительная учебная литература и справочная

1. Гуриков С. Р. Введение в программирование на языке Visual C# : учебное пособие по направлению "Инфокоммуникационные технологии и системы связи" / С. Р. Гуриков, 2013. - 444.

2. Введение в программирование и структуры данных / К. Фислер [и др.]; пер. с англ. А. В. Снастин, 2022. - 440.

9 Ресурсы сети Интернет

1. <http://library.istu.edu/>
2. <https://e.lanbook.com/>

10 Профессиональные базы данных

1. <http://new.fips.ru/>
2. <http://www1.fips.ru/>

11 Перечень информационных технологий, лицензионных и свободно распространяемых специализированных программных средств, информационных справочных систем

1. Лицензионное программное обеспечение Системное программное обеспечение
2. Лицензионное программное обеспечение Пакет прикладных офисных программ
3. Лицензионное программное обеспечение Интернет-браузер

12 Материально-техническое обеспечение дисциплины

1. Учебная аудитория для проведения лекционных занятий, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации. Оснащение: комплект учебной мебели, рабочее место преподавателя, доска. Мультимедийное оборудование (в том числе переносное): мультимедийный проектор, экран, акустическая система, компьютер с выходом в интернет.
2. Учебная аудитория для проведения лабораторных/практических (семинарских) занятий, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации. Оснащение: комплект учебной мебели, рабочее место преподавателя, доска. Мультимедийное оборудование (в том числе переносное): мультимедийный проектор, экран, акустическая система, компьютер с выходом в интернет.