

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

Структурное подразделение «Сибирская школа геонаук (119)»

**УТВЕРЖДЕНА:**  
на заседании ДЮТ  
Протокол №40 от 13 мая 2026 г.

**Рабочая программа дисциплины**  
**«ОСНОВЫ ПРОГРАММИРОВАНИЯ»**

---

Специальность: 21.05.03 Технология геологической разведки

---

Технология и техника разведки месторождений полезных ископаемых

---

Квалификация: Горный инженер-буровик

---

Форма обучения: очная

---

Документ подписан простой  
электронной подписью  
Составитель программы:  
Ланько Анна Викторовна  
Дата подписания: 08.06.2026

Документ подписан простой  
электронной подписью  
Утвердил: Ланько Анна  
Викторовна  
Дата подписания: 18.06.2026

Документ подписан простой  
электронной подписью  
Согласовал: Карпиков  
Александр Владимирович  
Дата подписания: 24.06.2026

Год набора – 2026

Иркутск, 2026 г.

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

## 1.1 Дисциплина «Основы программирования» обеспечивает формирование следующих компетенций с учётом индикаторов их достижения

Код, наименование компетенции	Код индикатора компетенции
ОПК-8 Способен применять основные методы, способы и средства получения, хранения и обработки информации, используя навыки работы с компьютером как средством управления информацией	ОПК-8.3

## 1.2 В результате освоения дисциплины у обучающихся должны быть сформированы

Код индикатора	Содержание индикатора	Результат обучения
ОПК-8.3	Способен применять основные методы, способы и средства получения, хранения и обработки информации в области программирования	<b>Знать</b> Понимать основы структур данных, алгоритмов и принципов работы программ для обработки геофизической информации. <b>Уметь</b> Уметь применять языки программирования и программные средства для обработки, хранения и анализа геофизических данных. <b>Владеть</b> Владеть навыками разработки, отладки и визуализации программ для автоматизации работы с геоинформацией.

## 2 Место дисциплины в структуре ООП

Изучение дисциплины «Основы программирования» базируется на результатах освоения следующих дисциплин/практик: «Информационные технологии»

Дисциплина является предшествующей для дисциплин/практик: «Проектная деятельность»

## 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 № 4
Общая трудоемкость дисциплины	144	144
Аудиторные занятия, в том числе:	65	65
лекции	26	26
лабораторные работы	39	39
практические/семинарские занятия	0	0
Самостоятельная работа (в т.ч. курсовое проектирование)	79	79

Трудоемкость промежуточной аттестации	0	0
Вид промежуточной аттестации (итогового контроля по дисциплине)	Зачет	Зачет

#### 4 Структура и содержание дисциплины

##### 4.1 Сводные данные по содержанию дисциплины

##### Семестр № 4

№ 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	1.Основы работы с Python	1	4	1	4			4	20	Отчет по лабораторной работе
2	2.Операторы и циклы	2	4	2	6					Отчет по лабораторной работе
3	3.Примитивы	3	4	3	6					Отчет по лабораторной работе
4	4.Функции	4	6	4, 5	12					Отчет по лабораторной работе
5	5.Базовые коллекции	5	4					2, 3	49	Отчет по лабораторной работе
6	6.Введение в ООП	6	4	6	11			1	10	Отчет по лабораторной работе
	Промежуточная аттестация									Зачет
	Всего		26		39				79	

##### 4.2 Краткое содержание разделов и тем занятий

##### Семестр № 4

№	Тема	Краткое содержание
1	1.Основы работы с Python	Общие сведения о языке программирования Python. Экскурс в историю языка. Положение языка программирования Python на рынке труда. Его преимущества и недостатки перед другими языками программирования высокого уровня. Краткое знакомство с синтаксисом языка, примеры реализации некоторых основных алгоритмов и сравнение реализаций на других языках программирования.
2	2.Операторы и циклы	Описание операторов в языке программирования Python. Оператор и операнд. 7 типов операторов: арифметические операторы, операторы сравнения,

		операторы присваивания, логические операторы, операторы принадлежности, операторы тождественности, битовые операторы
3	3.Примитивы	Описание примитивных типов данных: Численные типы данных: целые, вещественные и комплексные числа. Операции с ними. Строковый тип данных. Операции со строками
4	4.Функции	Определение функций в Python. Инструкции def, return, pass. Вызов функций, аргументы и параметры функций. Изменяемые аргументы по умолчанию. Позиционные аргументы. Документирование функций. Рекурсивные функции. Присвоение функции переменной.
5	5.Базовые коллекции	Описание коллекций в Python. Классификация коллекций. Свойства коллекций: индексированность, уникальность, изменяемость. Общие подходы при работе с коллекциями. Коллекции list, string, dict. Обход всех элементов коллекции в цикле for in. Функции min(), max(), sum(). Общие методы для части коллекций. Конвертации одного типа коллекции в другой. Получение, изменение значений по индексу, срезы коллекций.
6	6.Введение в ООП	Объектно-ориентированное программирование – концепция. Объектно-ориентированная парадигма. Понятие класс и экземпляр класса (объект). Методы класса. Динамическое изменение классов и методов. Статические методы. Жизненный цикл объекта – инициализаторы и деструкторы.

### 4.3 Перечень лабораторных работ

#### Семестр № 4

№	Наименование лабораторной работы	Кол-во академических часов
1	Ознакомление со средой разработки IDLE Python. Первая программа	4
2	Операторы и Циклы	6
3	Типы данных	6
4	Функции	6
5	Коллекции	6
6	Первые объектно-ориентированные программы	11

### 4.4 Перечень практических занятий

Практических занятий не предусмотрено

### 4.5 Самостоятельная работа

#### Семестр № 4

№	Вид СРС	Кол-во академических часов
1	Оформление отчетов по лабораторным и практическим работам	10
2	Подготовка к зачёту	29
3	Подготовка к практическим занятиям (лабораторным работам)	20
4	Проработка разделов теоретического материала	20

В ходе проведения занятий по дисциплине используются следующие интерактивные методы обучения: Работа в малых группах

## 5 Перечень учебно-методического обеспечения дисциплины

### 5.1 Методические указания для обучающихся по освоению дисциплины

#### 5.1.1 Методические указания для обучающихся по лабораторным работам:

Работа №1. Ознакомление со средой разработки IDLE Python. Первая программа

Цель: Познакомиться с интегрированной средой разработки IDLE Python, научиться создавать и запускать простейшие программы.

Ход выполнения:

- Запустить среду IDLE.
- Создать новый файл, написать первую программу (например, вывод текста на экран).
- Запустить и отладить программу.
- Ознакомиться с элементами интерфейса (окна ввода команд, редактор кода).

Результат: Рабочая программа, выведенный текст, понимание структуры программы и среды.

Контрольные вопросы: Что такое IDLE? Как создать и запустить программу в Python? Какие основные элементы кода содержатся в программе?

Работа №2. Операторы и циклы

Цель: Изучить операторы присваивания, арифметические и логические операторы, освоить циклы for и while.

Ход выполнения:

- Написать программу с использованием операторов присваивания и арифметики.
- Реализовать разные типы циклов для решения простых задач (например, вычисление суммы чисел).
- Освоить операторы ветвления внутри циклов.

Результат: Программы, использующие операторы и циклы.

Контрольные вопросы: Какие типы операторов бывают в Python? Как работают циклы for и while?

Работа №3. Типы данных

Цель: Познакомиться с основными типами данных в Python: числа, строки, логические значения, списки.

Ход выполнения:

- Создать переменные разных типов.
- Выполнить операции над ними (сложение, конкатенация, индексация).
- Ознакомиться с функциями преобразования типов.

Результат: Программы, демонстрирующие работу с разными типами данных.

Контрольные вопросы: Какие типы данных наиболее часто используются в Python? Чем отличаются списки от строк?

#### Работа №4. Функции

Цель: Изучить создание и использование функций в Python, понять передачу аргументов и возврат значений.

Ход выполнения:

- Написать функцию с параметрами, реализующую простую логику.
- Вызвать функцию из основного кода программы.
- Ознакомиться с областью видимости переменных.

Результат: Функциональный код, разделённый на логические блоки.

Контрольные вопросы: Как объявить функцию? Что такое параметры и возвращаемое значение?

#### Работа №5. Коллекции

Цель: Освоить работу с основными коллекциями Python: списки, кортежи, словари, множества.

Ход выполнения:

- Создать и заполнить разные коллекции.
- Выполнить операции добавления, удаления и поиска элементов.
- Использовать циклы для обхода коллекций.

Результат: Программы, работающие с коллекциями данных.

Контрольные вопросы: Чем отличаются списки от кортежей? Как обращаться к элементам словаря?

#### Работа №6. Первые объектно-ориентированные программы

Цель: Познакомиться с основами объектно-ориентированного программирования в Python: классы, объекты, методы, наследование.

Ход выполнения:

- Определить простой класс с атрибутами и методами.
- Создать объекты и вызвать методы.
- Реализовать наследование и переопределение методов.

Результат: Программы, демонстрирующие основные принципы ООП.

Контрольные вопросы: Что такое класс и объект? Как реализовать наследование в Python?

### 5.1.2 Методические указания для обучающихся по самостоятельной работе:

Подготовка к практическим занятиям (лабораторным работам):

Изучайте теоретический материал по теме занятия, просмотрите примеры кода и самостоятельно реализуйте простые программы для закрепления навыков.

Оформление отчетов по лабораторным и практическим работам:

Соблюдайте структуру отчета: цель, описание задания, ход выполнения, результаты, выводы; обязательно иллюстрируйте отчет фрагментами кода и комментариями.

Проработка разделов теоретического материала:

Повторяйте ключевые понятия, правила синтаксиса, алгоритмы; используйте конспекты, видео и дополнительные источники для глубокого понимания.

Подготовка к зачёту:

Систематизируйте знания в виде кратких заметок, решайте типовые задачи, тренируйтесь объяснять и применять изученные методы программирования устно и письменно.

## **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 4 | Отчет по лабораторной работе**

##### **Описание процедуры.**

Проект представляется в печатном и электронном варианте на формате А-4. Объем работы – не менее 25 страниц, записанном на диск в формате \*. doc и \*. pdf .

Титульный лист курсового проекта должен содержать тему работы, курс, группу, фамилию, инициалы автора, фамилию, должность (звание) научного руководителя.

##### **Критерии оценивания.**

Рейтинговая оценка курсовых работ осуществляется с применением критериев, аналогичных критериям оценки творческих работ, наряду с которыми целесообразно использовать такие критерии как:

- оригинальность работы;
- правильность и уместность использования информационного и методического аппарата (способов, методов, приемов, таблиц, графиков и пр.);
- правильность постановки и степень достижения поставленных задач;
- практическая значимость полученных результатов.

Примерные варианты распределения баллов по критериям оценки курсовых проектов представлены ниже (таблица 1). Конкретный вариант должен учитывать особенности тематики, по которой выполняется работа. При этом в нем должны быть учтены как минимум три критерия оценки.

Таблица 1 - Примерные варианты структуры оценки курсового проекта по критериям

№	Критерии оценки курсовых работ (проектов)	Баллы
1	Оформление работы	5
2	Умение искать необходимую информацию (литература)	10
3	Актуальность темы и оригинальность выполнения	10
4	Постановка и достижение цели	10
5	Правильность и уместность использования методов и информации	10
6	Практическая значимость полученных результатов	10
7	Логичность, умение обобщать, делать выводы	
8	Использование возможностей лабораторного оборудования, программного обеспечения и пр.	5
9	Защита курсовой работы	30
	Итоговый рейтинг по курсовой работе	100

### **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-8.3	полнота и правильность ответа; понимание и осознанность материала; логичность и последовательность изложения; корректность терминологии; способность отвечать на уточняющие вопросы	устное собеседование по теоретическим вопросам

## 6.2.2 Типовые оценочные средства промежуточной аттестации

### 6.2.2.1 Семестр 4, Типовые оценочные средства для проведения зачета по дисциплине

#### 6.2.2.1.1 Описание процедуры

Студент представляет программы, написанные в рамках лабораторных работ за семестр. Оценивается их полнота и качество. Задается два вопроса по алгоритмам и фрагментам кода случайно выбранных работ.

Примерные теоретические вопросы:

Возможности языка Python

- Загрузка и установка Python
- Первая программа. Знакомство со средой разработки IDLE
- Синтаксис
- Условный оператор if
- Циклы
- Ключевые слова, встроенные функции (для ознакомления, знать, как не стоит называть переменные)
- Числа
- Строки (часть 1, часть 2, форматирование)
- Списки (массивы)
- Кортежи
- Множества
- Функции
- Исключения и их обработка
- Байтовые строки
- Файлы
- Создание и подключение модулей
- Объектно-ориентированное программирование.

#### 6.2.2.1.2 Критерии оценивания

Зачтено	Не зачтено
Выполнены все следующие условия: - Все практические работы выполнены. - В результате вопросов по методике	Выполнено любое из следующих условий: - Часть практических работ не выполнена или выполнена с ошибками. - Не в

<p>выполнения практических работ установлено, что студент овладел практическими навыками, понимает программный код и алгоритмы (работы выполнены самостоятельно). - На теоретический вопрос дан удовлетворительный ответ.</p>	<p>состоянии объяснить методику выполнения работ, содержание кода, возможно самостоятельное их выполнение. Приемлемый ответ на теоретический вопрос не дан.</p>
---	---

## 7 Основная учебная литература

1. Носырева Е. В. Основы программирования [Электронный ресурс] : конспект лекций для специальности 230201 - "Информационные системы и технологии" / Е. В. Носырева, 2010. - 52.

[Сайт] – URL: <http://elib.istu.edu/viewer/view.php?file=/files/er-4365.pdf>

2. Силен Д. Основы Data Science и Big Data. Python и наука о данных / Д. Силен, А. Мейсман, М. Али; пер. с англ. Е. Матвеев, 2018. - 334.

3. Борзунов С. В. Алгебра и геометрия с примерами на Python : учебное пособие для вузов / С. В. Борзунов, С. Д. Кургалин, 2020. - 444.

[Сайт] – URL: <https://e.lanbook.com/book/149336>

4. Букунов С. В. Разработка приложений с графическим пользовательским интерфейсом на языке Python : учебное пособие / С. В. Букунов, О. В. Букунова, 2023. - 88.

[Сайт] – URL: <https://e.lanbook.com/book/292856>

## 8 Дополнительная учебная литература и справочная

1. Чернышев С. А. Основы программирования на Python : учебное пособие для вузов / С. А. Чернышев, 2024. - 349.

2. Бондарев Владимир Михайлович. Основы программирования / В. М. Бондарев, В. И. Рублинецкий, Е. Г. Качко, 1998. - 366.

3. Гриценко М. Б. Алгоритмические языки и основы программирования : пособие к практическим занятиям для специальности "Организация механизированной обработки экономической информации": для дневной и вечерней форм обучения / М. Б. Гриценко, 1977. - 36.

4. Рамальо Л. Python - к вершинам мастерства. Лаконичное и эффективное программирование / Л. Рамальо ; пер. А. А. Слинкин, 2022. - 898.

## 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.