Минобрнауки России

Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Оренбургский государственный университет имени В.А. Бондаренко»

Кафедра прикладной математики

РАБОЧАЯ ПРОГРАММА

ДИСЦИПЛИНЫ

«Б1.Д.Б.10 Информатика»

Уровень высшего образования

БАКАЛАВРИАТ

Направление подготовки
09.03.04 Программная инженерия
(код и наименование направления подготовки)

Разработка корпоративных систем и цифровых решений
(наименование направленности (профиля) образовательной программы)

Квалификация
Бакалавр

Форма обучения
Очная

Год набора 2026

Рабочая программа дисциплины «*Б1.Д.Б.10 Информатика*» рассмотрена и утверждена на заседании кафедры

Кафедра прикладной математики
наименование кафедры

протокол № _____от "___" _____ 20__г.

Заведующий кафедрой
Кафедра прикладной математики _____ И.П. Болодурина
наименование кафедры *подпись* *расшифровка подписи*

Исполнители:
Доцент каф. ПМ, к.т.н. _____ Л.С. Гришина
должность *подпись* *расшифровка подписи*

_____
должность *подпись* *расшифровка подписи*

СОГЛАСОВАНО:
Председатель методической комиссии по направлению подготовки
09.03.04 Программная инженерия _____ И.П. Болодурина
код наименование *личная подпись* *расшифровка подписи*

Заведующий отделом формирования фонда и научной обработки документов
_____ С.А. Бикмишерова
личная подпись *расшифровка подписи*

Уполномоченный по качеству института
_____ С.Н. Морозова
личная подпись *расшифровка подписи*

№ регистрации _____

© Гришина Л.С., 2026
© ОГУ, 2026

2

1 Цели и задачи освоения дисциплины

Цель (цели) освоения дисциплины:

Формирование у студентов фундаментальных представлений об архитектуре вычислительных систем, принципах алгоритмизации и современных подходах к разработке программного обеспечения на языке Python, а также развитие практических навыков создания модульного, объектно-ориентированного и профессионально оформленного кода для решения прикладных задач в области разработки корпоративных систем и цифровых решений.

Задачи:

- Освоение принципов функционирования вычислительных систем на основе архитектуры фон Неймана и понимание связи между аппаратным обеспечением, операционной системой и прикладным программным кодом.
- Формирование навыков алгоритмизации задач, оценки вычислительной сложности алгоритмов и выбора оптимальных структур данных для решения прикладных задач.
- Освоение синтаксиса и семантики языка Python, включая работу с встроенными типами данных, структурами данных (списки, словари, множества), функциями, модулями и объектно-ориентированной парадигмой.
- Развитие компетенций в области модульного программирования, обработки исключений, работы с файлами, регулярными выражениями и сериализацией данных.
- Освоение принципов создания графических пользовательских интерфейсов (GUI) и автоматизации рутинных задач с помощью настольных приложений.

Формирование навыков профессиональной разработки: тестирование, отладка, контроль качества кода, работа с системами контроля версий и инструментами автоматизации разработки.

2 Место дисциплины в структуре образовательной программы

Дисциплина относится к базовой части блока Д «Дисциплины (модули)»

Пререквизиты дисциплины: *Отсутствуют*

Постреквизиты дисциплины: *Б1.Д.Б.11 Основы программирования, Б1.Д.Б.14 Теория систем и системный анализ, Б1.Д.Б.17 Операционные системы, Б1.Д.Б.18 Вычислительные системы, сети и телекоммуникации, Б1.Д.Б.21 Основы проектной деятельности, Б1.Д.В.3 Системы управления базами данных*

3 Требования к результатам обучения по дисциплине

Процесс изучения дисциплины направлен на формирование следующих результатов обучения

Код и наименование формируемых компетенций	Код и наименование индикатора достижения компетенции	Планируемые результаты обучения по дисциплине, характеризующие этапы формирования компетенций
УК-1 Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	УК-1-В-4 Применяет методы сбора, хранения, обработки, передачи, анализа и синтеза информации с использованием компьютерных технологий для решения поставленных задач	**Знать:** Основные структуры данных Python (списки, словари, множества), алгоритмы поиска и сортировки, методы работы с файлами и сериализации данных, принципы обработки и анализа информации. **Уметь:** Применять встроенные коллекции Python для хранения и обработки данных, использовать алгоритмы поиска

Код и наименование формируемых компетенций	Код и наименование индикатора достижения компетенции	Планируемые результаты обучения по дисциплине, характеризующие этапы формирования компетенций
		и сортировки, работать с файлами различных форматов (CSV, JSON), применять регулярные выражения для парсинга текстовой информации. **Владеть:** Методами эффективной обработки данных с использованием стандартной библиотеки Python, инструментами сериализации и десериализации, подходами к валидации и очистке данных.
ОПК-2 Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности	ОПК-2-В-1 Понимает принципы работы современных информационных технологий и программных средств, в том числе отечественного производства ОПК-2-В-2 Использует современные информационные технологии в процессе разработки прикладных информационных систем в профессиональной сфере	**Знать:** Архитектуру вычислительных систем (принципы фон Неймана), модель памяти Python, принципы объектно-ориентированного программирования, механизмы инкапсуляции, наследования и полиморфизма, основы создания графических интерфейсов, принципы параллельных вычислений и асинхронного программирования. **Уметь:** Проектировать иерархии классов с использованием ООП, создавать настольные приложения с GUI, применять современные подходы к разработке, использовать инструменты разработки (IDE, системы контроля версий, виртуальные окружения). **Владеть:** Навыками объектно-ориентированного проектирования, создания GUI-приложений с использованием tkinter/PyQt, работы с виртуальными окружениями и системами контроля версий (Git), инструментами профессиональной разработки.
ОПК-8 Способен осуществлять поиск, хранение, обработку и анализ информации из различных источников и баз данных, представлять ее в требуемом формате с использованием информационных, компьютерных и сетевых технологий	ОПК-8-В-1 Осуществляет поиск, хранение, обработку и анализ информации с использованием баз данных	**Знать:** Основные форматы хранения и сериализации данных (CSV, JSON, YAML); базовые принципы организации баз данных. **Уметь:** Выполнять операции чтения и записи данных с использованием контекстных менеджеров; преобразовывать извлеченные из БД или файлов данные в структуры Python (списки, словари) для последующего алгоритмического анализа. **Владеть:** Практическими навыками программной работы с файловыми хранилищами

Код и наименование формируемых компетенций	Код и наименование индикатора достижения компетенции	Планируемые результаты обучения по дисциплине, характеризующие этапы формирования компетенций
		через стандартную библиотеку Python; приемами парсинга, очистки и структурирования «сырых» данных, извлеченных из внешних источников, для их дальнейшего анализа и представления в корпоративных приложениях.
ПК*-2 Способен разрабатывать и отлаживать программный код	ПК*-2-В-4 Разрабатывает программный код с использованием языков программирования ПК*-2-В-5 Оформляет программный код в соответствии с установленными требованиями	**Знать:** Синтаксис и семантику языка Python, принципы модульного программирования, механизмы обработки исключений, стандарты оформления кода (PEP 8), методы тестирования и отладки, инструменты контроля качества кода (линтеры, форматтеры, проверка типов). **Уметь:** Разрабатывать модульный код с использованием функций и классов, применять декораторы и замыкания, обрабатывать исключения, писать unit-тесты, использовать отладчик, применять линтеры и форматтеры для обеспечения качества кода. **Владеть:** Навыками написания чистого, читаемого и поддерживаемого кода, инструментами тестирования (pytest, unittest), профилирования и отладки, системами автоматизации проверки качества кода, принципами документирования кода.

4 Структура и содержание дисциплины

4.1 Структура дисциплины

Общая трудоемкость дисциплины составляет 3 зачетные единицы (108 академических часов).

Вид работы	Трудоемкость, академических часов	
	1 семестр	всего
Общая трудоёмкость	**108**	**108**
Контактная работа:	**52,25**	**52,25**
Лекции (Л)	18	18
Лабораторные работы (ЛР)	34	34
Промежуточная аттестация (зачет, экзамен)	0,25	0,25
Самостоятельная работа: - выполнение индивидуального творческого задания (ИТЗ); - самоподготовка (проработка и повторение лекционного материала и материала учебников и учебных пособий; - изучение разделов курса в системе электронного обучения;	**55,75**	**55,75**

Вид работы	Трудоемкость, академических часов	
	1 семестр	всего
- подготовка к лабораторным занятиям; - подготовка к рубежному контролю и т.п.)		
Вид итогового контроля (зачет, экзамен, дифференцированный зачет)	**зачет**	

Разделы дисциплины, изучаемые в 1 семестре

№ раздела	Наименование разделов	Количество часов				
		всего	аудиторная работа			внеауд. работа
			Л	ПЗ	ЛР	
1	Архитектура вычислительных систем и основы алгоритмизации	10	2		2	6
2	Базовый синтаксис и модель памяти языка Python	14	2		4	8
3	Встроенные структуры данных и алгоритмы обработки	12	2		4	6
4	Функциональное и модульное программирование	12	2		4	6
5	Объектно-ориентированное программирование	12	2		4	6
6	Работа с данными: файлы, сериализация, регулярные выражения	16	4		6	6
7	Разработка графических пользовательских интерфейсов (GUI)	16	2		6	8
8	Тестирование, отладка и профессиональные инструменты разработки	16	2		4	10
	Итого:	108	18		34	56
	Всего:	108	18		34	56

4.2 Содержание разделов дисциплины

1 Архитектура вычислительных систем и основы алгоритмизации.

Принципы фон Неймана. Уровни абстракции: от аппаратного обеспечения (CPU, RAM, кэш, шины) до прикладного ПО. Взаимодействие операционной системы и интерпретатора. Жизненный цикл разработки программного обеспечения (SDLC): этапы от постановки задачи до сопровождения. Теория алгоритмов: свойства алгоритма (дискретность, детерминированность, понятность, результативность, массовость). Правила построения блок-схем согласно ГОСТ 19.701-90. Вычислительная сложность алгоритмов. Нотация Big O. Анализ худшего, среднего и лучшего случая выполнения. Правила упрощения: отбрасывание констант, выделение старшего члена.

2 Базовый синтаксис и модель памяти языка Python.

Экосистема и инструментарий разработчика. Установка Python, Miniconda, виртуальные окружения. Обзор интегрированных сред разработки (VS Code, PyCharm, Jupyter). Системы контроля версий Git. Синтаксис языка и стандарт PEP 8. Идентификаторы, служебные слова, правила оформления отступов. Модель памяти Python: объекты, ссылки, куча (Heap). Пространства имён. Связь архитектуры ЭВМ с кодом: структура PyObject, счётчик ссылок, указатель на тип. Динамическая строгая типизация. Базовые скалярные типы данных и последовательности. Арифметические, логические операторы и операторы сравнения. Равенство и идентичность. Конструкции условного перехода if-elif-else, тернарный оператор. Циклы while и for, генераторы списков. Аннотации типов.

3 Встроенные структуры данных и алгоритмы обработки.

Коллекции в языке Python. Списки как динамические массивы, срезы, методы работы. Кортежи и строки: неизменяемость, кодировка Unicode. Словари: реализация на основе хеш-таблиц,

разрешение коллизий, сложность доступа O(1). Множества и frozenset: теория множеств, хешируемость объектов. Алгоритмы поиска: линейный поиск, бинарный поиск. Алгоритмы сортировки: квадратичные сортировки (пузырьком, выбором, вставками). Быстрая сортировка (Quick Sort), принцип «разделяй и властвуй». Встроенный алгоритм Timsort: стабильность, адаптивность. Сравнение производительности алгоритмов.

4 Функциональное и модульное программирование.

Функции и модульность. Принцип DRY. Позиционные и именованные аргументы, конструкции *args и **kwargs. Области видимости: правило LEGB. Замыкания. Анонимные функции (lambda). Декораторы. Рекурсия и мемоизация. Модули и пакеты: механизм импорта. Ключевые модули стандартной библиотеки. Обработка исключений: иерархия классов исключений. Блоки try-except-else-finally. Генерация исключений (raise).

5 Объектно-ориентированное программирование.

Философия и принципы ООП. Классы и объекты. Конструктор и инициализатор. Пространства имён и атрибуты. Четыре столпа ООП: инкапсуляция, наследование (MRO, алгоритм C3 Linearization), полиморфизм, абстракция. Магические методы: перегрузка операторов, эмуляция контейнеров, итераторов, контекстных менеджеров. Продвинутый ООП-подход: Data Classes. Структурная типизация (Protocols, PEP 544). Композиция против наследования. Архитектурные принципы SOLID.

6 Работа с данными: файлы, сериализация, регулярные выражения.

Файловые операции: текстовые и бинарные файлы. Контекстные менеджеры (with). Кодировки (UTF-8, ASCII). Чтение и запись данных в форматах CSV, JSON, YAML. Модули pathlib, csv, json, pickle. Регулярные выражения: модуль re. Синтаксис паттернов, метасимволы, группы, квантификаторы. Жадные и ленивые квантификаторы. Сериализация и десериализация данных. Форматы обмена данными. Библиотеки marshmallow и pydantic. Работа с API: модуль requests, HTTP-методы, заголовки, аутентификация.

7 Разработка графических пользовательских интерфейсов (GUI).

Принципы построения графических интерфейсов. Событийно-ориентированное программирование. Модель «источник события — обработчик». Основные паттерны взаимодействия с пользователем. Обзор библиотек: встроенная tkinter, современные альтернативы PyQt/PySide, CustomTkinter. Основные виджеты. Компоновка и размещение элементов. Менеджеры геометрии в tkinter (pack, grid, place). Сигналы и слоты в Qt. Привязка событий: обработка кликов, ввода текста, закрытия окна. Валидация пользовательского ввода. Автоматизация рутинных задач: интеграция GUI с файловыми операциями и обработкой данных. Упаковка приложений.

8 Тестирование, отладка и профессиональные инструменты разработки.

Тестирование программного кода. Пирамида тестов. Unit-тесты: модуль unittest. Фреймворк pytest: фикстуры, параметризация, маркеры. Покрытие кода. Отладка и профилирование. Профайлеры cProfile и timeit. Анализ памяти. Качество кода: линтеры (ruff, flake8), форматтеры (black, isort), проверка типов (mypy). Документирование кода. Упаковка и дистрибуция.

4.3 Лабораторные работы

№ ЛР	№ раздела	Наименование лабораторных работ	Кол-во часов
1	1	Конфигурирование среды разработки и установка IDE.	2
2	2	Базовые конструкции языка Python: типы данных, операторы и ввод-вывод.	2
3	2	Управляющие конструкции: ветвление и итерационные процессы	2
4	3	Встроенные структуры данных и обработка коллекций.	2
5	3	Алгоритмы поиска, сортировки и оценка их сложности.	2
6	4	Функции на Python.	2
7	4	Обработка исключений, генерация собственных исключений	2
8	5	Объектно-ориентированное проектирование: классы, наследование и полиморфизм.	4
9	6	Чтение и запись файлов, обработка текстов регулярными выражениями.	4

№ ЛР	№ раздела	Наименование лабораторных работ	Кол-во часов
10	6	Сериализация данных и взаимодействие с внешними API	2
11	7	Разработка графического пользовательского интерфейса настольного приложения.	4
12	7	Автоматизация задач средствами GUI.	2
13	8	Модульное тестирование и контроль качества кода.	4
		Итого:	34

5 Учебно-методическое обеспечение дисциплины

5.1 Основная литература

1. Информатика [Текст] : учебное пособие / под ред. Б. Е. Одинцова, А. Н. Романова.- 2- е изд., перераб. и доп. - Москва : Вузовский учебник : ИНФРА-М, 2012. - 410 с. : ил. - (Вузовский учебник). - Библиогр.: с. 404-405. - ISBN 978-5-9558-0230-5. - ISBN 978-5-16-005108-6.1.

2. Степанов, А. Н. Информатика [Текст] : учеб. пособие для вузов / А. Н. Степанов.- 5- е изд. - Санкт Петербург : Питер, 2008. - 765 с. : ил. - (Учебник для вузов). - Библиогр.: с. 754. - Алф. указ.: с. 755. - ISBN 978-5-469-01348-8.

3. Информатика. Базовый курс [Текст] : учеб. пособие для студентов втузов: для бакалавров и специалистов / под ред. С. В. Симоновича.- 3-е изд. - СПб. : Питер, 2012. - 638 с. : ил. - (Учебник для вузов) - ISBN 978-5-459-00439-7.

4. Сузи, Р. А. Язык программирования Python [Текст] : учеб. пособие / Р. А. Сузи. - М. : ИНТУИТ. РУ : БИНОМ. Лаборатория знаний, 2006. - 326 с. : ил. - (Основы информационных технологий). - Библиогр.: с. 325-326. - ISBN 5-9556-0058-2. - ISBN 5-94774-442-2.

5.2 Дополнительная литература

1. Елович, И. В. Информатика [Текст] : учебник для студентов высших учебных заведений, обучающихся по техническим и естественно-научным направлениям / И. В. Елович, И. В. Кулибаба; под ред. Г. Г. Раннева. - Москва : Академия, 2011. - 395 с. : ил. - (Высшее профессиональное образование. Бакалавриат). - Библиогр.: с. 388-390. - ISBN 978-5-7695-7975-2

2. Острейковский, В. А. Информатика: учебник / В. А. Острейковский. - М. : Высш. шк., 2001. - 511 с. : ил - ISBN 5-06-003533-6.

3. Акулов, О. А. Информатика [Текст] : базовый курс: учеб. для вузов, бакалавров, магистров, обучающихся по направлению "Информатика и вычисл. техника" / О. А. Акулов, Н. В. Медведев; [гл. ред. Г. Л. Гуртова].- 5-е изд., испр. и доп. - Москва : Омега - Л, 2008. - 574 с. : ил. - (Высшее техническое образование). - Библиогр.: с. 573-574. - ISBN 978-5-365-00901-1.

4. Сузи, Р. А. Язык программирования Python [Текст] : учеб. пособие / Р. А. Сузи. - М. : ИНТУИТ. РУ : БИНОМ. Лаборатория знаний, 2006. - 326 с. : ил. - (Основы информационных технологий). - Библиогр.: с. 325-326. - ISBN 5-9556-0058-2. - ISBN 5-94774-442-2.

5. Основы программирования на языке Python [Текст]: учебное пособие / С. К. Буйначев, Н. Ю. Боклаг. – Екатеринбург: Изд-во Урал. ун-та, 2014. – 91 с. – ISBN 978-5-7996-1198-9. – URL: https://elar.urfu.ru/bitstream/10995/28769/1/978-5-7996-1198-9_2014.pdf (дата обращения: 20.07.2026). – Режим доступа: свободный.

6. Ханжин, С. В. Основы программирования на Python [Электронный ресурс] : учебное пособие для обучающихся по образовательной программе высшего образования по направлениям подготовки 12.03.04 Биотехнические системы и технологии, 03.03.03 Радиофизика / С. В. Ханжин; М-во науки и высш. образования Рос. Федерации, Федер. гос. бюджет. образоват. учреждение высш. образования "Оренбург. гос. ун-т им. В. А. Бондаренко". - Электрон. текстовые дан. (1 файл: 1.29 Мб). - Оренбург : ОГУ, 2026. - 124 с. - Загл. с тит. экрана. - Adobe Acrobat Reader 6.0. - Режим доступа: http://artlib.osu.ru/web/books/metod_all/240547_20260428.pdf - ISBN 978-5-7410-3600-6.

5.3 Периодические издания

Информатика и системы управления : журнал. - М. : Агентство "Роспечать", 2026.

2. Информационные технологии в проектировании и производстве : журнал. - М. : Агентство "Роспечать", 2026.

3. Мир ПК : журнал. - М. : Агентство "Роспечать", 2026.

4. Программные продукты и системы : журнал. - М. : Агентство "Роспечать", 2026.

5.4 Интернет-ресурсы

1. http://www.citforum.ru/ – портал аналитических и научных статей в области информационных технологий;

2. http://www.soft.cnews.ru/ – Новости в сфере информационных технологий, обзоры нового программного обеспечения, статьи, результаты тестирования новых программных продуктов.

5.5 Программное обеспечение, профессиональные базы данных и информационные справочные системы

1 Операционная система РЕД ОС для рабочих станций, имеется лицензия, входит в реестр отечественного ПО. – Режим доступа: https://redos.red-soft.ru/

2 LibreOffice – свободно распространяемый офисный пакет программ, включающий в себя текстовый и табличный редакторы, редактор презентаций и другие офисные приложения. – Режим доступа: https://ru.libreoffice.org/ 9

3 Программа для просмотра сайтов Яндекс.Браузер, свободно распространяемая, входит в реестр отечественного ПО. – Режим доступа: https://yandex.ru/

4 Программная система для организации видео-конференц-связи DION. – Режим доступа: https://diongo.ru/

5 Университетская платформа электронного обучения «Электронные курсы ОГУ в системе обучения Moodle». – Режим доступа: http://moodle.osu.ru

6 Автоматизированная интерактивная система сетевого тестирования (АИССТ). – Режим доступа: https://aist.osu.ru/cgi-bin/auth.cgi

7 Elibrary [Электронный ресурс]: реферативная база данных, с ограниченным доступом к полным текстам статей. – Режим доступа: https://www.elibrary.ru/ , в локальной сети ОГУ.

8 Python 3.13+ [Электронный ресурс]: интерпретатор языка программирования Python. — Режим доступа: https://www.python.org/, свободно.

9 Интегрированная среда разработки ПО Visual Studio Code, свободно распространяемая по лицензии MIT. — Режим доступа: https://code.visualstudio.com/.

10 PyCharm Community Edition [Электронный ресурс]: интегрированная среда разработки. / Разработчик JetBrains. — Режим доступа: https://www.jetbrains.com/pycharm/, свободно.

11 GitVerse [Электронный ресурс]: отечественная платформа для совместной разработки и управления версиями программного кода. / Разработчик ООО «Гитверс». — Режим доступа: https://gitverse.ru/, бесплатно для открытых репозиториев и учебных команд разработчиков.

6 Материально-техническое обеспечение дисциплины

Учебные аудитории для проведения занятий лекционного типа, семинарского типа, для проведения групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации. Для чтения лекций используется переносной мультимедийный комплект: ноутбук, проектор, экран.

Аудитории оснащены комплектами ученической мебели, техническими средствами обучения, служащими для представления учебной информации большой аудитории.

Для проведения лабораторных занятий используется компьютерный класс, оснащенный оснащенный компьютерами с минимальными характеристиками: оперативная память: не менее 8 Gb; процессор не менее чем на 4 ядра и частотой не менее 2,5 Ghz; объём памяти видеокарты не менее 2

Gb; жесткий диск не менее чем на 500 Gb; наличие USB-разъема на лицевой стороне системного блока; диагональ ЖК монитора не менее 21.5".

Помещение для самостоятельной работы обучающихся оснащены компьютерной техникой, подключенной к сети "Интернет", и обеспечением доступа в электронную информационно-образовательную среду ОГУ.