

Министерство науки и высшего образования Российской Федерации
Муромский институт (филиал)
федерального государственного бюджетного образовательного учреждения высшего образования
**«Владимирский государственный университет
имени Александра Григорьевича и Николая Григорьевича Столетовых»**
(МИ ВлГУ)

Кафедра *ФПМ*

«УТВЕРЖДАЮ»
Заместитель директора по УР
Д.Е. Андрианов
_____ 19.05.2026

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Технологии и методы программирования

Направление подготовки

01.03.02 Прикладная математика и информатика

Профиль подготовки

Интеллектуальный анализ данных

Семестр	Трудоем- кость, час./зач. ед.	Лек- ции, час.	Практи- ческие занятия, час.	Лабора- торные работы, час.	Консуль- тация, час.	Конт- роль, час.	Всего (контак- тная работа), час.	СРС, час.	Форма промежу- точного контроля (экз., зач., зач. с оц.)
1	108 / 3	30	16		5	0,35	51,35	21	Экз.(35,65)
3	108 / 3	32	16		3,2	0,25	51,45	56,55	Зач. с оц.
4	72 / 2	32	16		3,2	2,25	53,45	18,55	Зач. с оц.
5	108 / 3	32	18		3,2	0,25	53,45	54,55	Зач. с оц.
6	72 / 2	32	16		3,2	0,25	51,45	20,55	Зач. с оц.
Итого	468 / 13	158	82		17,8	3,35	261,15	171,2	35,65

Муром, 2026 г.

1. Цель освоения дисциплины

Целью изучения дисциплины является освоение современных технологий и подходов в программировании.

Задачи:

- изучение таких принципов объектно-ориентированного программирования как инкапсуляция, наследование, полиморфизм;
- освоение приемов работы в современных средах программирования;
- получение навыков программирования с использованием современных языков программирования;
- создание приложений, использующих собственные и стандартные библиотеки классов;
- использование средств групповой разработки программ и их отладки;
- использование средств поддержки разработки программ.

2. Место дисциплины в структуре ОПОП ВО

Изучение дисциплины базируется на дисциплинах: «Информатика», «Математика», «Теория вероятностей и математическая статистика», «Дискретная математика». Дисциплина обеспечивает изучение дисциплин: «Управление данными», «Проектирование информационных систем». Знания и практические навыки, полученные в рамках изучения дисциплины «Технологии и методы программирования», используются при разработке курсовых и выпускных квалификационных работ.

3. Планируемые результаты обучения по дисциплине

Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения ОПОП (компетенциями и индикаторами достижения компетенций)

Формируемые компетенции (код, содержание компетенции)	Планируемые результаты обучения по дисциплине, в соответствии с индикатором достижения компетенции		Наименование оценочного средства
	Индикатор достижения компетенции	Результаты обучения по дисциплине	
ПК-1 Способность выполнять интеграцию программных модулей и компонент и верификацию выпусков программного продукта	ПК-1.1 Разрабатывает, тестирует и интегрирует программные модули и компоненты	Знать технологии интеграции и верификации программного обеспечения (ПК-1.1) Уметь выполнять разработку, интеграцию программных модулей и компонент и верификацию выпусков программного продукта (ПК-1.1) Владеть технологиями разработки, интеграции и верификации программного обеспечения (ПК-1.1)	отчет, тест

4. Структура и содержание дисциплины

Общая трудоемкость дисциплины составляет 13 зачетных единиц, 468 часов.

4.1. Форма обучения: очная

Уровень базового образования: среднее общее.

Срок обучения 4г.

4.1.1. Структура дисциплины

№ п/п	Раздел (тема) дисциплины	Семестр	Контактная работа обучающихся с педагогическим работником							Самостоятельная работа	Форма текущего контроля успеваемости (по неделям семестра), форма промежуточной аттестации(по семестрам)
			Лекции	Практические занятия	Лабораторные работы	Контрольные работы	КП / КР	Консультация	Контроль		
1	Основы Python	1	30	16						21	практическая работа, тестирование
Всего за семестр		108	30	16				5	0,35	21	Экз.(35,65)
2	Современный стандарт C++	3	32	16						56,55	практическая работа, тестирование
Всего за семестр		108	32	16				3,2	0,25	56,55	Зач. с оц.
3	Современный стандарт C++ ч.2	4	32	16						18,55	практическая работа, тестирование
Всего за семестр		72	32	16			+	3,2	2,25	18,55	Зач. с оц.
4	Программирование на языке Java	5	32	18						54,55	практическая работа, тестирование
Всего за семестр		108	32	18				3,2	0,25	54,55	Зач. с оц.
5	Модули Python (библиотеки)	6	32	16						20,55	практическая работа, тестирование
Всего за семестр		72	32	16				3,2	0,25	20,55	Зач. с оц.
Итого		468	158	82				17,8	3,35	171,2	35,65

4.1.2. Содержание дисциплины

4.1.2.1. Перечень лекций

Семестр 1

Раздел 1. Основы Python

Лекция 1.

Введение в Python: синтаксис, типы данных, переменные (2 часа).

Лекция 2.

Условные операторы (if-elif-else) (2 часа).

Лекция 3.

Циклы (for, while) (2 часа).

Лекция 4.

Функции (определение, аргументы, возврат значений) (2 часа).

Лекция 5.

Списки, кортежи, множества (2 часа).

Лекция 6.

Словари и работа с ними (2 часа).

Лекция 7.

Работа со строками (str, методы, форматирование) (2 часа).

Лекция 8.

Файлы: чтение и запись (open, with) (2 часа).

Лекция 9.

Обработка исключений (try-except-else-finally) (2 часа).

Лекция 10.

Модули и импорт (import, pip) (2 часа).

Лекция 11.

Основы ООП в Python (классы, объекты, методы) (2 часа).

Лекция 12.

Наследование и полиморфизм (2 часа).

Лекция 13.

Декораторы и генераторы (2 часа).

Лекция 14.

Работа с датами и временем (datetime) (2 часа).

Лекция 15.

Введение в numpy и простые операции с массивами (2 часа).

Семестр 3**Раздел 2. Современный стандарт C++****Лекция 16.**

Введение в C++11: auto, nullptr, range-based for (2 часа).

Лекция 17.

Умные указатели (unique_ptr, shared_ptr, weak_ptr) (2 часа).

Лекция 18.

STL: контейнеры (vector, list, deque) (2 часа).

Лекция 19.

STL: итераторы и алгоритмы (sort, find, transform) (2 часа).

Лекция 20.

Лямбда-функции и функциональные объекты (2 часа).

Лекция 21.

STL: ассоциативные контейнеры (map, set, unordered_map) (2 часа).

Лекция 22.

Многопоточность (std::thread, std::async) (2 часа).

Лекция 23.

Базовые алгоритмы STL (поиск, сортировка, модификация) (2 часа).

Лекция 24.

Работа с файлами (fstream, filesystem в C++17) (2 часа).

Лекция 25.

Обработка ошибок и исключения (try-catch, noexcept) (2 часа).

Лекция 26.

Время и дата (chrono) (2 часа).

Лекция 27.

Регулярные выражения (std::regex) (2 часа).

Лекция 28.

Метапрограммирование: constexpr, if constexpr (2 часа).

Лекция 29.

Введение в move-семантику (rvalue-ссылки) (2 часа).

Лекция 30.

Параллельные алгоритмы (C++17) (2 часа).

Лекция 31.

Практика: реализация простого проекта с STL (2 часа).

Семестр 4

Раздел 3. Современный стандарт C++ ч.2

Лекция 32.

Углубленное изучение move-семантики и perfect forwarding (2 часа).

Лекция 33.

Шаблоны и SFINAE (std::enable_if) (2 часа).

Лекция 34.

Введение в концепты (C++20) (2 часа).

Лекция 35.

STL: специализированные контейнеры (stack, queue, priority_queue) (2 часа).

Лекция 36.

STL: аллокаторы и пользовательские allocators (2 часа).

Лекция 37.

Многопоточность: мьютексы, условные переменные (2 часа).

Лекция 38.

Атомарные операции и std::atomic (2 часа).

Лекция 39.

Сетевые возможности (boost.asio или std::net из C++23) (2 часа).

Лекция 40.

Сериализация (JSON: nlohmann/json, бинарная сериализация) (2 часа).

Лекция 41.

STL: строки и string_view (2 часа).

Лекция 42.

Оптимизация производительности (профилирование, кеширование) (2 часа).

Лекция 43.

Метапрограммирование: std::variant, std::visit (2 часа).

Лекция 44.

Многопоточные паттерны (thread pool, producer-consumer) (2 часа).

Лекция 45.

Работа с графикой (SFML / OpenGL обертка) (2 часа).

Лекция 46.

Введение в COROUTINES (C++20) (2 часа).

Лекция 47.

Финальный проект: многопоточный сервер с STL (2 часа).

Семестр 5

Раздел 4. Программирование на языке Java

Лекция 48.

Введение в Java: JVM, синтаксис, типы данных (2 часа).

Лекция 49.

Классы и объекты, методы, конструкторы (2 часа).

Лекция 50.

Наследование, интерфейсы, абстрактные классы (2 часа).

Лекция 51.

Коллекции: List, Set, Map (2 часа).

Лекция 52.

Stream API (Java 8): map, filter, reduce (2 часа).

Лекция 53.

Многопоточность: Thread, Runnable, ExecutorService (2 часа).

Лекция 54.

Обработка исключений (try-catch-finally) (2 часа).

Лекция 55.

Работа с файлами (Files, Path, BufferedReader) (2 часа).

Лекция 56.

Аннотации и рефлексия (2 часа).

Лекция 57.

Лямбда-выражения и функциональные интерфейсы (2 часа).

Лекция 58.

Работа с датами (java.time) (2 часа).

Лекция 59.

Сериализация (JSON: Jackson, Gson) (2 часа).

Лекция 60.

Сетевое программирование (Socket, HTTP-клиенты) (2 часа).

Лекция 61.

Базы данных: JDBC, Hibernate (начало) (2 часа).

Лекция 62.

Юнит-тестирование (JUnit) (2 часа).

Лекция 63.

Финальный проект: REST-сервис на Spring Boot (2 часа).

Семестр 6

Раздел 5. Модули Python (библиотеки)

Лекция 64.

NumPy: массивы, операции, индексация (2 часа).

Лекция 65.

Pandas: DataFrame, Series, группировки (2 часа).

Лекция 66.

Визуализация (matplotlib, seaborn) (2 часа).

Лекция 67.

Работа с базами данных (sqlite3, SQLAlchemy) (2 часа).

Лекция 68.

Парсинг данных (BeautifulSoup, requests) (2 часа).

Лекция 69.

Многопоточность (threading, multiprocessing) (2 часа).

Лекция 70.

Асинхронность (asyncio, aiohttp) (2 часа).

Лекция 71.

Работа с API (REST, FastAPI) (2 часа).

Лекция 72.

Тестирование (unittest, pytest) (2 часа).

Лекция 73.

Логирование (logging) (2 часа).

Лекция 74.

Обработка изображений (Pillow, OpenCV) (2 часа).

Лекция 75.

Машинное обучение (scikit-learn – основы) (2 часа).

Лекция 76.

NLP (nltk, spacy – начало) (2 часа).

Лекция 77.

Веб-скрапинг (Scrapy) (2 часа).

Лекция 78.

Автоматизация (selenium) (2 часа).

Лекция 79.

Финальный проект: Data Pipeline (сбор + анализ + визуализация) (2 часа).

4.1.2.2. Перечень практических занятий

Семестр 1

Раздел 1. Основы Python

Практическое занятие 1

Базовые операции: калькулятор с вводом через `input()` (2 часа).

Практическое занятие 2

Условные операторы: проверка числа на четность/нечетность, определение високосного года (2 часа).

Практическое занятие 3

Циклы: вывод таблицы умножения, поиск простых чисел (2 часа).

Практическое занятие 4

Функции: вычисление факториала, числа Фибоначчи (2 часа).

Практическое занятие 5

Работа со списками: сортировка, фильтрация, обработка (2 часа).

Практическое занятие 6

Словари: частотный анализ текста (подсчет слов) (2 часа).

Практическое занятие 7

Файлы: чтение/запись CSV, обработка данных (2 часа).

Практическое занятие 8

Мини-проект: консольный "Терминал" (меню с вызовом функций) (2 часа).

Семестр 3

Раздел 2. Современный стандарт C++

Практическое занятие 9

`vector`: реализация динамического массива с сортировкой (2 часа).

Практическое занятие 10

`map/set`: словарь синонимов (ввод-поиск) (2 часа).

Практическое занятие 11

Умные указатели: имитация работы `shared_ptr` (2 часа).

Практическое занятие 12

STL-алгоритмы: обработка данных (например, статистика по тексту) (2 часа).

Практическое занятие 13

Многопоточность: параллельное вычисление суммы элементов массива (2 часа).

Практическое занятие 14

`filesystem`: обход директории и поиск файлов по маске (2 часа).

Практическое занятие 15

Лямбды: сортировка объектов по произвольному полю (2 часа).

Практическое занятие 16

Мини-проект: "Агрегатор новостей" (парсинг и хранение в `vector<Article>`) (2 часа).

Семестр 4

Раздел 3. Современный стандарт C++ ч.2

Практическое занятие 17

Move-семантика: оптимизация кода с `std::move` (2 часа).

Практическое занятие 18

Контейнеры STL: сравнение производительности `vector`, `list`, `deque` (2 часа).

Практическое занятие 19

Многопоточность: реализация паттерна `Producer-Consumer` (2 часа).

Практическое занятие 20

Сериализация: сохранение/загрузка объектов в JSON (2 часа).

Практическое занятие 21

Сети: простой HTTP-клиент (запрос к API) (2 часа).

Практическое занятие 22

`string_view`: оптимизация обработки строк (2 часа).

Практическое занятие 23

Асинхронность: загрузка файлов через `std::async` (2 часа).

Практическое занятие 24

Финальный проект: чат-сервер с многопоточной обработкой (2 часа).

Семестр 5

Раздел 4. Программирование на языке Java

Практическое занятие 25

Классы и объекты: модель "Студент-Группа" (2 часа).

Практическое занятие 26

Коллекции: рейтинговая книга (сортировка по баллам) (2 часа).

Практическое занятие 27

Stream API: фильтрация и агрегация данных (например, статистика по товарам) (2 часа).

Практическое занятие 28

Многопоточность: параллельный расчет чисел Фибоначчи (2 часа).

Практическое занятие 29

Файлы: парсинг CSV и преобразование в объекты (2 часа).

Практическое занятие 30

JDBC: подключение к БД, выполнение запросов (2 часа).

Практическое занятие 31

Юнит-тесты: покрытие кода для класса "Калькулятор" (2 часа).

Практическое занятие 32

Spring Boot: создание REST-эндпоинта (GET/POST) (2 часа).

Практическое занятие 33

Мини-проект: "Учет книг в библиотеке" (CRUD + интерфейс) (2 часа).

Семестр 6

Раздел 5. Модули Python (библиотеки)

Практическое занятие 34

NumPy: операции с матрицами (транспонирование, умножение) (2 часа).

Практическое занятие 35

Pandas: анализ датасета (средние, мода, корреляции) (2 часа).

Практическое занятие 36

Matplotlib: визуализация данных (графики, гистограммы) (2 часа).

Практическое занятие 37

Парсинг: сбор данных с веб-страницы (например, курсы валют) (2 часа).

Практическое занятие 38

Асинхронность: скачивание нескольких URL через `aiohttp` (2 часа).

Практическое занятие 39

FastAPI: создание API для управления задачами (TODO-лист) (2 часа).

Практическое занятие 40

Scikit-learn: предсказание на основе линейной регрессии (2 часа).

Практическое занятие 41

Финальный проект: "Анализатор логов" (сбор + визуализация статистики) (2 часа).

4.1.2.3. Перечень лабораторных работ

Не планируется.

4.1.2.4. Перечень тем и учебно-методическое обеспечение самостоятельной работы

Перечень тем, вынесенных на самостоятельное изучение:

1. Типы данных и переменные в Python. Примеры использования.
2. Операции с числами в Python. Примеры использования.
3. Условные операторы в Python. Примеры использования.
4. Циклы в Python. Примеры использования.

5. Работа со списками и строками в Python. Примеры использования.
6. Основные методы программирования. Сравнительный анализ языков программирования.
7. Методы ООП. Инкапсуляция. С++. Вопросы практического применения.
8. Методы ООП. Наследование. С++. Вопросы практического применения.
9. Методы ООП. Полиморфизм. С++. Вопросы практического применения.
10. Современный стандарт С++. Вопросы практического применения.
11. Разработка Java приложений.
12. Обзор Android Studio.
13. Разработка Java приложений для системы Android.
14. Использование библиотеки NumPy. Примеры использования.
15. Использование библиотеки Matplotlib. Примеры использования.
16. Использование шаблонов проектирования. Примеры использования.
17. Технологии коллективной разработки программного обеспечения. Примеры использования.
18. Использование CASE-средств разработки программного обеспечения. Примеры использования.
19. Методы и средства отладки и тестирования программ. Примеры использования.

Для самостоятельной работы используются методические указания по освоению дисциплины и издания из списка приведенной ниже основной и дополнительной литературы.

4.1.2.5. Перечень тем контрольных работ, рефератов, ТР, РГР, РПР

Не планируется.

4.1.2.6. Примерный перечень тем курсовых работ (проектов)

1. Разработка программы, сравнивающей содержимое двух текстовых файлов.
2. Разработка простейшего растрового редактора.
3. Разработка программы моделирования электромагнитных колебаний.
4. Разработка игровой программы «Пятнашки».
5. Разработка программы печати этикеток для компакт-дисков.
6. Разработка программы моделирования движения пружинного маятника.
7. Разработка программы мониторинга молокоотдачи.
8. Разработка программы расчета разветвленной цепи постоянного тока.
9. Разработка программы-калькулятора логических выражений.
10. Разработка программы «Программируемый калькулятор».
11. Разработка программы моделирования баллистического движения.

5. Образовательные технологии

В рамках изучения дисциплины используются интерактивные технологии преподавания, выраженные в виде совместных обсуждений проблемных ситуаций, совместного анализа путей решения поставленных задач. В рамках выполнения практических работ формируются небольшие коллективы из студентов для совместного решения задач. Результаты работы отдельных коллективов обсуждаются всей группой, при этом используются средства мультимедийной техники. Преподаватель выступает в роли координатора работы коллективов студентов, дает оценку их работе.

6. Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины.

Фонды оценочных материалов (средств) приведены в приложении.

7. Учебно-методическое и информационное обеспечение дисциплины.

7.1. Основная учебно-методическая литература по дисциплине

1. Программирование на языке Python: Практикум для студентов образовательной программы 09.03.02 Информационные системы и технологии / сост. Терехин А.В. [Электронный ресурс]. – Электрон. текстовые дан. (11 Мб). - Муром: МИ ВлГУ, 2017. - 1 электрон. опт. диск (CD-R). – Систем. требования: процессор x86 с тактовой частотой 500 МГц и выше; 512 Мб ОЗУ; Windows XP/7/8; видеокарта SVGA 1280x1024 High Color (32 bit); привод CD-ROM. - Загл. с экрана. - <https://evrika.mivlgu.ru/admin/index.php?r=books/view&id=3026>
2. Сузи, Р. А. Язык программирования Python : учебное пособие / Р. А. Сузи. — 3-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2020. — 350 с. — ISBN 978-5-4497-0705-5. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <http://www.iprbookshop.ru/97589.html> - <http://www.iprbookshop.ru/97589.html>
3. Маккинли, Уэс Python и анализ данных / Уэс Маккинли ; перевод А. Слинкина. — 2-е изд. — Саратов : Профобразование, 2019. — 482 с. — ISBN 978-5-4488-0046-7. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <http://www.iprbookshop.ru/88752.html> - <http://www.iprbookshop.ru/88752.html>
4. Рик, Гаско Простой Python просто с нуля / Гаско Рик. — Москва : СОЛОН-Пресс, 2019. — 256 с. — ISBN 978-5-91359-334-4. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <http://www.iprbookshop.ru/94940.html> - <http://www.iprbookshop.ru/94940.html>
5. Шелудько, В. М. Основы программирования на языке высокого уровня Python : учебное пособие / В. М. Шелудько. — Ростов-на-Дону, Таганрог : Издательство Южного федерального университета, 2017. — 146 с. — ISBN 978-5-9275-2649-9. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <http://www.iprbookshop.ru/87461.html> - <http://www.iprbookshop.ru/87461.html>
6. Шелудько, В. М. Язык программирования высокого уровня Python. Функции, структуры данных, дополнительные модули : учебное пособие / В. М. Шелудько. — Ростов-на-Дону, Таганрог : Издательство Южного федерального университета, 2017. — 107 с. — ISBN 978-5-9275-2648-2. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <http://www.iprbookshop.ru/87530.html> - <http://www.iprbookshop.ru/87530.html>
7. Буйначев, С. К. Основы программирования на языке Python : учебное пособие / С. К. Буйначев, Н. Ю. Боклаг ; под редакцией Ю. В. Песин. — Екатеринбург : Уральский федеральный университет, ЭБС АСВ, 2014. — 92 с. — ISBN 978-5-7996-1198-9. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <http://www.iprbookshop.ru/66183.html> - <http://www.iprbookshop.ru/66183.html>
8. Смирнов, А. А. Технологии программирования : учебное пособие / А. А. Смирнов, Д. В. Хрипков. — Москва : Евразийский открытый институт, 2011. — 191 с. — ISBN 978-5-374-00296-6. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <http://www.iprbookshop.ru/10900.html> - <http://www.iprbookshop.ru/10900.html>
9. Васильев, В. Н. Основы программирования на языке C# : учебное пособие / В. Н. Васильев. — Волгоград : Волгоградский институт бизнеса, 2010. — 70 с. — ISBN 978-5-9061-7234-1. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <http://www.iprbookshop.ru/11341.html> - <http://www.iprbookshop.ru/11341.html>
10. Агапов, В. П. Основы программирования на языке C# : учебное пособие / В. П. Агапов. — Москва : Московский государственный строительный университет, ЭБС АСВ, 2012. — 128 с. — ISBN 978-5-7264-0576-6. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <http://www.iprbookshop.ru/16366.html> - <http://www.iprbookshop.ru/16366.html>
11. Кивран, В. К. Программирование в среде Visual C++ 6 : учебное пособие / В. К. Кивран. — Самара : Самарский государственный архитектурно-строительный университет, ЭБС АСВ, 2014. — 118 с. — ISBN 978-5-9585-0601-9. — Текст : электронный // Электронно-

библиотечная система IPR BOOKS : [сайт]. — URL: <http://www.iprbookshop.ru/43185.html> - <http://www.iprbookshop.ru/43185.html>

12. Лисицин, Д. В. Объектно-ориентированное программирование : конспект лекций / Д. В. Лисицин. — Новосибирск : Новосибирский государственный технический университет, 2010. — 88 с. — ISBN 978-5-7782-1454-5. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <http://www.iprbookshop.ru/44970.html> - <http://www.iprbookshop.ru/44970.html>

13. Программирование на языке высокого уровня C/C++ : конспект лекций / составители С. П. Зоткин. — Москва : Московский государственный строительный университет, Ай Пи Эр Медиа, ЭБС АСВ, 2016. — 140 с. — ISBN 978-5-7264-1285-6. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <http://www.iprbookshop.ru/48037.html> - <http://www.iprbookshop.ru/48037.html>

14. Букунов, С. В. Основы программирования на языке C++ : учебное пособие / С. В. Букунов. — Санкт-Петербург : Санкт-Петербургский государственный архитектурно-строительный университет, ЭБС АСВ, 2015. — 201 с. — ISBN 978-5-9227-0619-3. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <http://www.iprbookshop.ru/63631.html> - <http://www.iprbookshop.ru/63631.html>

15. Котов, О. М. Язык C#. Краткое описание и введение в технологии программирования : учебное пособие / О. М. Котов. — Екатеринбург : Уральский федеральный университет, ЭБС АСВ, 2014. — 208 с. — ISBN 978-5-7996-1094-4. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <http://www.iprbookshop.ru/68524.html> - <http://www.iprbookshop.ru/68524.html>

16. Медведев, М. А. Программирование на СИ# : учебное пособие / М. А. Медведев, А. Н. Медведев ; под редакцией А. В. Присяжный. — Екатеринбург : Уральский федеральный университет, ЭБС АСВ, 2015. — 64 с. — ISBN 978-5-7996-1561-1. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <http://www.iprbookshop.ru/69667.html> - <http://www.iprbookshop.ru/69667.html>

17. Биллиг, В. А. Основы объектного программирования на C# (C# 3.0, Visual Studio 2008) : учебник / В. А. Биллиг. — 3-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2021. — 409 с. — ISBN 978-5-4497-0880-9. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <http://www.iprbookshop.ru/102029.html> - <http://www.iprbookshop.ru/102029.html>

18. Белева, Л. Ф. Программирование на языке C++ : учебное пособие / Л. Ф. Белева. — Саратов : Ай Пи Эр Медиа, 2018. — 81 с. — ISBN 978-5-4486-0253-5. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <http://www.iprbookshop.ru/72466.html> - <http://www.iprbookshop.ru/72466.html>

19. Вязовик, Н. А. Программирование на Java : учебное пособие / Н. А. Вязовик. — 3-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2021. — 601 с. — ISBN 978-5-4497-0852-6. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <http://www.iprbookshop.ru/102048.html> - <http://www.iprbookshop.ru/102048.html>

20. Павловская, Т. А. Программирование на языке высокого уровня C# : учебное пособие / Т. А. Павловская. — 3-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2021. — 245 с. — ISBN 978-5-4497-0862-5. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <http://www.iprbookshop.ru/102051.html> - <http://www.iprbookshop.ru/102051.html>

21. Страуструп, Б. Язык программирования C++ для профессионалов : учебник / Б. Страуструп. — 3-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2021. — 670 с. — ISBN 978-5-4497-0922-6. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <http://www.iprbookshop.ru/102077.html> - <http://www.iprbookshop.ru/102077.html>

22. Фридман, А. Л. Язык программирования C++ : учебное пособие / А. Л. Фридман. — 3-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2021. — 217 с. — ISBN 978-5-4497-0920-2. — Текст : электронный // Электронно-

библиотечная система IPR BOOKS : [сайт]. — URL: <http://www.iprbookshop.ru/102076.html> - <http://www.iprbookshop.ru/102076.html>

7.2. Дополнительная учебно-методическая литература по дисциплине

1. Журнал "RSDN MAGAZINE" - <http://elibrary.ru/contents.asp?titleid=10139>
2. Журнал "ТРУДЫ ИНСТИТУТА СИСТЕМНОГО ПРОГРАММИРОВАНИЯ РАН" - <http://elibrary.ru/contents.asp?titleid=31961>
3. Журнал "ПРОГРАММНЫЕ СИСТЕМЫ: ТЕОРИЯ И ПРИЛОЖЕНИЯ" - <http://elibrary.ru/contents.asp?issueid=1387854>
4. Журнал "ПРОГРАММНЫЕ ПРОДУКТЫ И СИСТЕМЫ" - <http://swsys.ru/>
5. Журнал "ВЕСТНИК ЮЖНО-УРАЛЬСКОГО ГОСУДАРСТВЕННОГО УНИВЕРСИТЕТА. СЕРИЯ: МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ И ПРОГРАММИРОВАНИЕ" - <http://elibrary.ru/contents.asp?titleid=26854>

7.3. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения и информационных справочных систем

В образовательном процессе используются информационные технологии, реализованные на основе информационно-образовательного портала института (www.mivlgu.ru/iop), и инфокоммуникационной сети института:

- предоставление учебно-методических материалов в электронном виде;
- взаимодействие участников образовательного процесса через локальную сеть института и Интернет;
- предоставление сведений о результатах учебной деятельности в электронном личном кабинете обучающегося.

Информационные справочные системы:

- Microsoft Developer Network - <https://msdn.microsoft.com/ru-ru/default.aspx>

- Официальный ресурс разработчиков Python - <https://www.python.org/doc/>

Программное обеспечение:

Google Chrome (Лицензионное соглашение Google)

РЕД ОС (Соглашение №140/05-21У от 18.05.2021 года о сотрудничестве в области науки, развития инновационной деятельности)

FireBird (Initial Developer's Public License и InterBase Public Licence)

Pycharm Community Edition (проприетарная лицензия и Apache License 2.0)

Python 3.9.4 (Python Software Foundation License)

7.4. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

evrika.mivlgu.ru

iprbookshop.ru

elibrary.ru

swsys.ru

python.org

mivlgu.ru/iop

8. Материально-техническое обеспечение дисциплины

Компьютерный класс

ПК CPU-Intel Core i5-4460 BOX - 12 шт.; ПК — 1шт.; экран DRAPPER Apex STAR; видеопроектор InFocus; коммутатор. Доступ к сети Интернет.

Лекционная аудитория

Проектор ViewSonic PG603X DLP Экран Cactus Wallscreen

Компьютерный класс
Проектор ViewSonic PG603X DLP Экран Lumien Персональный компьютер RUSCO –
19 шт. Коммутатор D-Link Маршрутизатор беспроводной N ASUS RT-AC66U

9. Методические указания по освоению дисциплины

Для успешного освоения теоретического материала обучающийся: ведет конспект лекций и прорабатывает лекционный материал, пользуясь как конспектом, так и учебными пособиями; знакомится со списком рекомендуемой основной и дополнительной литературы; уточняет у преподавателя, какими дополнительными учебными пособиями следует пользоваться.

Практические занятия используются для закрепления теоретического материала, решения практических задач с целью формирования профессиональных умений и навыков. Занятия проводятся в компьютерном классе с использованием специального программного обеспечения. Задания на практические занятия выбираются обучающимися из методических указаний согласно индивидуальному варианту. В конце занятия обучающие демонстрируют полученные результаты преподавателю, при необходимости делают работу над ошибками.

Самостоятельная работа призвана обеспечить углубленное изучение вопросов, рассматриваемых во время аудиторных занятий. Самостоятельная работа обучающихся строится на основе установленного перечня тем для самостоятельного изучения. Каждый обучающийся самостоятельно определяет режим своей работы и меру труда, затрачиваемого на овладение учебным содержанием дисциплины. Он выполняет внеаудиторную работу и изучение разделов, выносимых на самостоятельную работу, по личному индивидуальному плану, в зависимости от собственного уровня подготовленности, времени и других условий.

Курсовая работа выполняется в соответствии с методическими указаниями на курсовую работу. Обучающийся выбирает одну из указанных в перечне тем курсовых работ, исходя из своих интересов, наличия соответствующих литературных и иных источников. В ходе выполнения курсовой работы преподаватель проводит консультации обучающегося. На заключительном этапе обучающийся оформляет пояснительную записку к курсовой работе и выполняет ее защиту в присутствии комиссии из преподавателей кафедры.

Форма заключительного контроля при промежуточной аттестации – зачет с оценкой. Для проведения промежуточной аттестации по дисциплине разработаны фонд оценочных средств и балльно-рейтинговая система оценки учебной деятельности студентов. Оценка по дисциплине выставляется в информационной системе и носит интегрированный характер, учитывающий результаты оценивания участия студентов в аудиторных занятиях, качества и своевременности выполнения заданий в ходе изучения дисциплины и промежуточной аттестации.

Программа составлена в соответствии с требованиями ФГОС ВО по направлению *01.03.02 Прикладная математика и информатика* и профилю подготовки *Интеллектуальный анализ данных*

Рабочую программу составил *д.т.н. Орлов А.А.* _____

Программа рассмотрена и одобрена на заседании кафедры *ФПМ*

протокол № 18 от 18.03.2026 года.

Заведующий кафедрой *ФПМ* _____ *Орлов А.А.*

(Подпись)

Рабочая программа рассмотрена и одобрена на заседании учебно-методической комиссии факультета

протокол № 9 от 12.05.2026 года.

Председатель комиссии *ФИТР* _____ *Кутарова Е.И.*

(Подпись)

(Ф.И.О.)

Фонд оценочных материалов (средств) по дисциплине
Технологии и методы программирования

**1. Оценочные материалы для проведения текущего контроля успеваемости
по дисциплине**

Задания для выполнения на практических занятиях

1. Описать структуру с именем PRICE, содержащую следующие поля:

- название товара;
- название магазина, в котором продается товар;
- стоимость товара в рублях.

Написать программу, выполняющую следующие действия:

- ввод с клавиатуры данных в массив, состоящий из восьми элементов типа PRICE (записи должны быть упорядочены в алфавитном порядке по названиям магазинов);
- вывод на экран информации о товарах, продающихся в магазине, название которого введено с клавиатуры (если такого магазина нет, вывести соответствующее сообщение).

2. Описать структуру с именем ORDER, содержащую следующие поля:

- расчетный счет плательщика;
- расчетный счет получателя;
- перечисляемая сумма в рублях.

Написать программу, выполняющую следующие действия:

- ввод с клавиатуры данных в массив, состоящий из восьми элементов типа ORDER (записи должны быть размещены в алфавитном порядке по расчетным счетам плательщиков);
- вывод на экран информации о сумме, снятой с расчетного счета плательщика, введенного с клавиатуры (если такого расчетного счета нет, вывести соответствующее сообщение).

3. Составить описание класса для представления комплексных чисел. Обеспечить выполнение операций сложения, вычитания и умножения комплексных чисел. Написать программу, демонстрирующую все разработанные элементы класса.

4. Составить описание класса для вектора, заданного координатами его концов в трехмерном пространстве. Обеспечить операции сложения и вычитания векторов с получением нового вектора (суммы или разности), вычисления скалярного произведения двух векторов, длины вектора, косинуса угла между векторами. Написать программу, демонстрирующую все разработанные элементы класса.

5. Составить описание класса прямоугольников со сторонами, параллельными осям координат. Предусмотреть возможность перемещения прямоугольников на плоскости, изменение размеров, построение наименьшего прямоугольника, содержащего два заданных прямоугольниками прямоугольника, являющегося общей частью (пересечением) двух прямоугольников. Написать программу, демонстрирующую все разработанные элементы класса.

6. Составить описание класса для представления даты. Предусмотреть возможности установки даты и изменения ее отдельных полей (год, месяц, день) с проверкой допустимости вводимых значений. В случае недопустимых значений полей выбрасываются исключения. Создать методы изменения даты на заданное количество дней, месяцев и лет. Написать программу, демонстрирующую все разработанные элементы класса.

7. Составить описание класса для представления времени. Предусмотреть возможности установки времени и изменения его отдельных полей (час, минута, секунда) с проверкой допустимости вводимых значений. В случае недопустимых значений полей выбрасываются исключения. Создать методы изменения времени на заданное количество

часов, минут и секунд. Написать программу, демонстрирующую все разработанные элементы класса.

8. Описать класс, представляющий треугольник. Предусмотреть методы для создания объектов, вычисления площади, периметра и точки пересечения медиан. Описать свойства для получения состояния объекта. При невозможности построения треугольника выбрасывается исключение. Написать программу, демонстрирующую все разработанные элементы класса.

9. Описать класс, представляющий круг. Предусмотреть методы для создания объектов, вычисления площади круга, длины окружности и проверки попадания заданной точки внутрь круга. Описать свойства для получения состояния объекта. Написать программу, демонстрирующую все разработанные элементы класса.

10. Описать класс для работы со строкой, позволяющей хранить только двоичное число и выполнять с ним арифметические операции. Предусмотреть инициализацию с проверкой допустимости значений. В случае недопустимых значений выбрасываются исключения. Написать программу, демонстрирующую все разработанные элементы класса.

11. В одномерном массиве, состоящем из n вещественных элементов, вычислить:

- сумму отрицательных элементов массива;
- произведение элементов массива, расположенных между максимальным и минимальным элементами.

Упорядочить элементы массива по возрастанию.

12. В одномерном массиве, состоящем из n вещественных элементов, вычислить:

- сумму положительных элементов массива;
- произведение элементов массива, расположенных между максимальным по модулю и минимальным по модулю элементами.

Упорядочить элементы массива по убыванию.

13. В одномерном массиве, состоящем из n вещественных элементов, вычислить:

- минимальный элемент массива;
- сумму элементов массива, расположенных между первым и последним положительными элементами.

Преобразовать массив таким образом, чтобы сначала располагались все элементы, равные нулю, а потом — все остальные.

14. В одномерном массиве, состоящем из n вещественных элементов, вычислить:

- номер минимального элемента массива;
- сумму элементов массива, расположенных между первым и вторым отрицательными элементами.

Преобразовать массив таким образом, чтобы сначала располагались все элементы, модуль которых не превышает единицу, а потом — все остальные.

15. Дана целочисленная прямоугольная матрица. Определить:

- количество столбцов, содержащих хотя бы один нулевой элемент;
- номер строки, в которой находится самая длинная серия одинаковых элементов.

16. Дана целочисленная квадратная матрица. Определить:

- произведение элементов в тех строках, которые не содержат отрицательных элементов;
- максимум среди сумм элементов диагоналей, параллельных главной диагонали матрицы.

17. Дана целочисленная квадратная матрица. Определить:

- сумму элементов в тех столбцах, которые не содержат отрицательных элементов;
- минимум среди сумм модулей элементов диагоналей, параллельных побочной диагонали матрицы.

18. Для заданной матрицы размером 8×8 найти такие k , при которых k -я строка матрицы совпадает с k -м столбцом. Найти сумму элементов в тех строках, которые содержат хотя бы один отрицательный элемент.

19. Осуществить циклический сдвиг элементов прямоугольной матрицы на n элементов вправо или вниз (в зависимости от введенного режима), n может быть больше количества элементов в строке или столбце.

20. Упорядочить строки целочисленной прямоугольной матрицы по возрастанию количества одинаковых элементов в каждой строке. Найти номер первого из столбцов, не содержащих ни одного отрицательного элемента.

21. Дана целочисленная прямоугольная матрица. Определить:

- количество строк, содержащих хотя бы один нулевой элемент;
- номер столбца, в котором находится самая длинная серия одинаковых элементов.

22. Дана целочисленная квадратная матрица. Определить:

- сумму элементов в тех строках, которые не содержат отрицательных элементов;
- минимум среди сумм элементов диагоналей, параллельных главной диагонали матрицы.

23. Описать класс «запись», содержащий следующие закрытые поля:

- фамилия, имя;
- номер телефона;
- дата рождения (массив из трех чисел).

Предусмотреть свойства для получения состояния объекта.

Описать класс «записная книжка», содержащий закрытый массив записей. Обеспечить следующие возможности:

- вывод на экран информации о человеке, номер телефона которого введен с клавиатуры; если такого нет, выдать на дисплей соответствующее сообщение;
- поиск людей, день рождения которых сегодня или в заданный день;
- поиск людей, день рождения которых будет на следующей неделе;
- поиск людей, номер телефона которых начинается на три заданных цифры.

Написать программу, демонстрирующую все разработанные элементы классов.

24. Создать класс Point (точка). На его основе создать классы ColoredPoint и Line (линия). На основе класса Line создать классы ColoredLine и PolyLine (многоугольник). В классах описать следующие элементы:

- конструкторы с параметрами и конструкторы по умолчанию;
- свойства для установки и получения значений всех координат, а также для изменения цвета и получения текущего цвета;
- для линий — методы изменения угла поворота линий относительно первой точки;
- для многоугольника — метод масштабирования.

25. Создать абстрактный класс Vehicle (транспортное средство). На его основе реализовать классы Plane (самолет), Car (автомобиль) и Ship (корабль). Классы должны иметь возможность задавать и получать координаты и параметры средств передвижения (цена, скорость, год выпуска и т. п.) с помощью свойств. Для самолета должна быть определена высота, для самолета и корабля — количество пассажиров, для корабля — порт приписки. Динамические характеристики задать с помощью методов.

Общее распределение баллов текущего контроля по видам учебных работ для студентов

Рейтинг-контроль 1	5 практических работ (3 семестр), 5 практических работ (4 семестр), 5 практических работ (5 семестр), 5 практических работ (6 семестр), 5 практических работ (7 семестр)	10 (1 семестр), 20 (3 семестр), 20 (4 семестр), 20 (5 семестр), 20 (6 семестр)
Рейтинг-контроль 2	5 практических работ (3	20 (1 семестр), 40 (3

	семестр), 6 практических работ (4 семестр), 6 практических работ (5 семестр), 6 практических работ (6 семестр), 6 практических работ (7 семестр)	семестр), 40 (4 семестр), 40 (5 семестр), 40 (6 семестр)
Рейтинг-контроль 3	5 практических работ (3 семестр), 5 практических работ (4 семестр), 5 практических работ (5 семестр), 5 практических работ (6 семестр), 5 практических работ (7 семестр)	20 (1 семестр), 30 (3 семестр), 30 (4 семестр), 30 (5 семестр), 30 (6 семестр)
Посещение занятий студентом		0 (1 семестр), 0 (3 семестр), 0 (4 семестр), 0 (5 семестр), 0 (6 семестр)
Дополнительные баллы (бонусы)		0 (1 семестр), 0 (3 семестр), 0 (4 семестр), 0 (5 семестр), 0 (6 семестр)
Выполнение семестрового плана самостоятельной работы		10 (1 семестр), 10 (3 семестр), 10 (4 семестр), 10 (5 семестр), 10 (6 семестр)

2. Промежуточная аттестация по дисциплине

Перечень вопросов к экзамену / зачету / зачету с оценкой.

Перечень практических задач / заданий к экзамену / зачету / зачету с оценкой (при наличии)

Теоретические, вопросы в форме теста и практические задания для проведения промежуточной аттестации приведены далее (в разделе 3).

Методические материалы, характеризующие процедуры оценивания

На основе типовых заданий программным комплексом информационно-образовательного портала МИ ВлГУ формируются в автоматическом режиме тестовые задания для студентов. Программный комплекс формирует индивидуальные задания для каждого зарегистрированного в системе студента и устанавливает время прохождения тестирования. Результатом тестирования является процент правильных ответов, с учетом индивидуального семестрового рейтинга студента формируется экзаменационная оценка.

Максимальная сумма баллов, набираемая студентом по дисциплине равна 100.

Оценка в баллах	Оценка по шкале	Обоснование	<i>Уровень сформированности компетенций</i>
Более 80	«Отлично»	Содержание курса освоено полностью, без пробелов, необходимые практические навыки работы с освоенным материалом сформированы, все предусмотренные программой	<i>Высокий уровень</i>

		обучения учебные задания выполнены, качество их выполнения оценено числом баллов, близким к максимальному	
66-80	«Хорошо»	Содержание курса освоено полностью, без пробелов, некоторые практические навыки работы с освоенным материалом сформированы недостаточно, все предусмотренные программой обучения учебные задания выполнены, качество выполнения ни одного из них не оценено минимальным числом баллов, некоторые виды заданий выполнены с ошибками	<i>Продвинутый уровень</i>
50-65	«Удовлетворительно»	Содержание курса освоено частично, но пробелы не носят существенного характера, необходимые практические навыки работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий, возможно, содержат ошибки	<i>Пороговый уровень</i>
Менее 50	«Неудовлетворительно»	Содержание курса не освоено, необходимые практические навыки работы не сформированы, выполненные учебные задания содержат грубые ошибки	<i>Компетенции не сформированы</i>

3. Задания в тестовой форме по дисциплине

Примеры заданий:

В каких случаях целесообразно использовать рекурсию?

Выберите один ответ:

- a. Когда применение циклов приводит к сильному усложнению кода программы
 - b. Рекурсию лучше избегать
 - c. Когда задачу нельзя решить с использованием циклов
- Хэш функция это

Выберите один ответ:

- a. Преобразование массива входных данных в битовую строку
- b. Функция предназначенная для кодирования или шифрования информации
- c. функция, осуществляющая преобразование массива входных данных произвольной длины в (выходную) битовую строку установленной длины, выполняемое определённым алгоритмом.

Асимптотическая сложность алгоритма это

Выберите один ответ:

- 1. функция определяющая число итераций

2. понятие в информатике и теории алгоритмов, обозначающее функцию зависимости объёма работы, которая выполняется некоторым алгоритмом, от размера входных данных.

3. функция, позволяющая определить, сколько времени потребуется на решение задачи
В скольких направлениях можно перемещаться в односвязном списке?

Выберите один ответ:

a. В двух

b. В одном

Очередь это:

Выберите один ответ:

a. Структура данных работающая по принципу FILO

b. Структура данных, работающая по принципу FIFO

c. Это двусторонний линейный список

Полный перечень тестовых заданий с указанием правильных ответов, размещен в банке вопросов на информационно-образовательном портале института по ссылке <https://www.mivlgu.ru/iop/question/edit.php?courseid=2629&category=30541%2C80817&qshowtext=0&recurse=0&recurse=1&showhidden=0>

<https://www.mivlgu.ru/iop/question/edit.php?courseid=397&cat=39397%2C10257>

<https://www.mivlgu.ru/iop/question/edit.php?courseid=402&cat=2031%2C10539>

Оценка рассчитывается как процент правильно выполненных тестовых заданий из их общего числа.