

## АННОТАЦИЯ

### дисциплины Б1.О.14 ТЕХНОЛОГИЯ ПРОГРАММИРОВАНИЯ И РАБОТА НА ЭЛЕКТРОННО-ВЫЧИСЛИТЕЛЬНОЙ МАШИНЕ (ЭВМ)

**Объем трудоемкости:** 16 зачетных единиц (576 часов, из них – 352 часа аудиторной нагрузки: лекционных 136 ч, практических 216 ч, 200,8 ч самостоятельной работы, 22 ч КСР, 1 ч ИКР)

**Цель дисциплины:** подготовка в области применения современной вычислительной техники для практического решения прикладных и теоретических математических задач.

**Задачи дисциплины:**

- ознакомить студентов с основными структурами данных и методами структурного программирования, современными информационными технологиями и системами программирования;
- научить применять современные информационные технологии для решения различных задач;
- обучить основам программирования на алгоритмических языках высокого уровня;
- научить использовать методы математического и алгоритмического моделирования при решении теоретических и прикладных математических задач.

**Место дисциплины в структуре ООП ВПО**

Дисциплина «Технология программирования и работы на ЭВМ» относится к базовой части Блока 1 "Дисциплины (модули)" учебного плана.

**Требования к уровню освоения дисциплины**

Изучение данной учебной дисциплины направлено на формирование у обучающихся следующих компетенций:

Код и наименование индикатора достижения компетенции	Результаты обучения по дисциплине
<b>УК-1 Способен осуществлять критический анализ проблемных ситуаций на основе системного подхода, вырабатывать стратегию действий</b>	
ИУК-1.2. Осуществляет поиск, отбор и систематизацию информации для определения альтернативных вариантов стратегических решений в проблемной ситуации и обоснования выбора оптимальной стратегии с учетом поставленной цели, рисков и возможных последствий	<p>Знает основные подходы к алгоритмизации решения математических и практических задач</p> <p>Умеет разрабатывать, анализировать, адаптировать и совершенствовать алгоритмы</p> <p>Владеет навыками практического использования математических алгоритмов</p>
<b>ПК-4 способен разрабатывать программное обеспечение для решения прикладных задач в сфере профессиональной деятельности</b>	
ИПК-1.2. Имеет навыки использования современных языков программирования для разработки программного обеспечения	<p>Знает основные приемы используемые при разработке программного обеспечения на современных языках программирования</p> <p>Умеет реализовывать алгоритмы с использованием современных языков программирования</p> <p>Владеет навыками практического программирования</p>
<b>ОПК-3 Способен понимать принципы работы современных информационных технологий и использовать их для решения задач профессиональной деятельности</b>	
ИОПК-3.1. Имеет представление о принципах работы современных информационных технологий	<p>Знает базовые принципы работы современных информационных технологий, основные методы математического и алгоритмического моделирования</p> <p>Умеет выбирать подходящие программные и технические средства для решения конкретных задач</p> <p>Владеет базовыми инструментами, используемыми в современных информационных технологиях</p>
<b>ОПК-5 Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения</b>	
ИОПК-5.2. Реализует алгоритмы с	Знает основные концепции, используемые при

Код и наименование индикатора достижения компетенции	Результаты обучения по дисциплине
использованием современных средств разработки прикладного программного обеспечения	программной реализации алгоритмов
	Умеет выбирать корректные методы при программной реализации решения математических задач
	Владеет навыками применения современных средств разработки прикладного программного обеспечения

### Основные разделы дисциплины:

№	Наименование раздела (темы)	Содержание раздела (темы)
1.	Алгоритмы и структуры данных	Классификация ЯП. Знакомство с Python. Типы данных и переменные. Базовые операции. Интерпретация ошибок. PEP8. Алгоритмизация. Языки программирования и алгоритмы. Математические множества и типы данных. Структуры данных: множества, связанные списки, стеки, дек, очереди, словари, хэш-таблицы, двоичные и префиксные деревья, графы.
2.	Функции алгебры логики	Функции алгебры логики. Логический тип. Операции отношения. Логические операции. Условные операторы. Составной оператор. Метки. Отладка программ в Python.
3.	Функции	Функции. Исключения и их обработка. Именованные функции, инструкция def. Аргументы функции. Анонимные функции, инструкция lambda. Рекурсия. Прямой и обратный ход рекурсии.
4.	Массивы, циклы и сортировки	Массивы. Операторы цикла. Списки, кортежи и словари. Операторы общие для всех типов последовательностей. Строки. Итераторы, генераторы, генераторы списков. Индексы и срезы. Методы сортировки. Пузырьковая сортировка и её улучшения. Сортировка перемешиванием (шейкерная сортировка). Сортировка расчёской. Сортировка вставками. Сортировка выбором.
5.	Конечные автоматы	Детерминированные и недетерминированные конечные автоматы. Таблица переходов. Конечные автоматы с магазинной памятью.
6.	Строки	Срезы строк и списков. Присваивание в срез списка. Стандартные функции len, max, min, sum. Методы split и join. Структура данных Куча/Пирамида (Heap) Расстояние Левенштейна. Проверка равенства строк. Наивный поиск подстроки в строке. Алгоритм Кнута-Морриса-Пратта.
7.	Модули и пакеты в Python	Основные стандартные модули и пакеты в Python. Импорт модулей. Создание собственных модулей и их импорт. Специализированные модули и приложения.
8.	Файлы	Ввод-вывод информации. Система управления вводом-выводом. Методы доступа. Метки файлов. Открытие и закрытие файлов. Логические и физические файлы. Общие операции над файлами. Типизированные файлы. Бестиповые файлы. Файлы прямого доступа. Текстовые файлы.
9.	Работа с графикой	Библиотека Matplotlib. Контейнер Figure. Область рисования. Мультиоконные рисунки. Координатные оси. Деления координатных осей. Специальные элементы рисунка в matplotlib. Особенности координатных осей. Графики в полярной системе координат. Реализация собственных функций.
10.	Библиотека для анализа	Создание массивов NumPy. Арифметические операции над массивами NumPy.

	данных NumPy	Индексация массива NumPy. Агрегирование в NumPy. Операции над матрицами и векторами в NumPy. Индексация матрицы NumPy. Агрегирование матриц NumPy. Транспонирование и изменение формы матриц в NumPy. Таблицы NumPy. Аудио и временные ряды в NumPy. Обработка изображений и текста в NumPy.
11.	Библиотека для анализа данных pandas.	Библиотека pandas. Работа с Датафреймами (таблицами) в pandas. Фильтрация наблюдений. Группировка и агрегирование. Графики и pandas. Визуализация. Регулярные выражения.
12.	Объектно-ориентированное программирование. Классы и методы	Преимущества и недостатки ООП. Классы. Объекты. Атрибуты класса. Атрибуты класса против атрибутов экземпляров. Методы. Статические методы. Возврат множественных значений из метода. Метод str. Конструкторы. Локальные и глобальные переменные. Модификаторы доступа.
13.	Наследование, полиморфизм и инкапсуляция	Наследование. Абстрактные классы. Уровень абстракции. Множественное наследование. Полиморфизм. Перегрузка метода. Переопределение метода. Инкапсуляция. Аксессуары (геттеры и сеттеры). Интерфейсы.
14.	Основы построения параллельных вычислительных систем	Архитектура параллельных вычислительных систем. Статическая и динамическая конвейеризация. Векторные вычисления. Систематика Флинна. Примеры параллельных вычислений. Параллелизм. Ускорение. Эффективность. Масштабируемость. Оценка эффективности разработки. Ограничения сети. Методы параллелизма. Процессы и потоки. Типичные ошибки многопоточного программирования. Взаимное исключение и условная синхронизация. Модель памяти и низкоуровневые примитивы синхронизации. Альтернативные подходы к реализации одновременных программ.
15.	Параллельные вычисления	Области применения и проблематика. Современные параллельные вычислительные системы. Теоретические основы параллельных вычислений. Показатели качества параллельного алгоритма. Принципы разработки и типовые структуры параллельных алгоритмов. Методология PCAM. Системы параллельного программирования. Типовые модели программирования и шаблоны. Основы параллельного программирования на системах с общей памятью на примере технологии OpenMP. Основы параллельного программирования на системах с распределенной памятью на примере технологии MPI.
16.	Параллельная обработка больших массивов данных	Феномен Big Data. Модель программирования MapReduce. Принципы параллельной реализации вычислений. Область применения и примеры задач. Принципы распределенной реализации MapReduce на кластерных системах. Платформа Apache Hadoop. Интерфейсы прикладного программирования и реализация программ для Hadoop. Локальная отладка и запуск программ на кластере. Приемы и стратегии реализации MapReduce-программ. Высокоуровневые языки и инструментарии для работы с Hadoop. Практические примеры использования MapReduce. Ограничения модели MapReduce, расширения и альтернативные подходы.
17.	Распределенные системы и вычисления	Области применения, характерные особенности и виды распределенных систем. Проблемы построения распределенных систем. Теоретические основы распределенных вычислений, примеры распределенных алгоритмов. Способы взаимодействия распределенных процессов, сетевые протоколы.
18.	Технологии распределенного программирования	Технологии распределенного программирования. Распределенные системы хранения данных, репликация данных, NoSQL-системы. Технологии распределенных вычислений, гриды, добровольные вычисления. Облачные вычислительные системы.

Изучение дисциплины заканчивается аттестацией в форме зачета.

### **Учебная литература:**

1. Окулов, С. М. Программирование в алгоритмах: учебное пособие / С. М. Окулов. — 7-е изд. — Москва: Лаборатория знаний, 2021. — 386 с. — ISBN 978-5-93208-521-9. — Текст: электронный // Лань: электронно-библиотечная система. — URL: <https://e.lanbook.com/book/172252>
2. Окулов, С. М. Алгоритмы обработки строк: учебное пособие / С. М. Окулов. — 4-е изд. — Москва: Лаборатория знаний, 2020. — 258 с. — ISBN 978-5-00101-658-8. — Текст: электронный // Лань: электронно-библиотечная система. — URL: <https://e.lanbook.com/book/135553>
3. Федоров, Д. Ю. Программирование на языке высокого уровня Python: учебное пособие для вузов / Д. Ю. Федоров. — 3-е изд., перераб. и доп. — Москва: Издательство Юрайт, 2021. — 210 с. — (Высшее образование). — ISBN 978-5-534-14638-7. — Текст: электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/478098>
4. Зыков, С. В. Программирование. Функциональный подход: учебник и практикум для вузов / С. В. Зыков. — Москва: Издательство Юрайт, 2021. — 164 с. — (Высшее образование). — ISBN 978-5-534-00844-9. — Текст: электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/470387>
5. Зыков, С. В. Программирование. Объектно-ориентированный подход: учебник и практикум для вузов / С. В. Зыков. — Москва: Издательство Юрайт, 2021. — 155 с. — (Высшее образование). — ISBN 978-5-534-00850-0. — Текст: электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/470281>