

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Ястребов Олег Александрович

Должность: Ректор

Дата подписания: 28.05.2026 10:28:14

Уникальный программный ключ:

ca953a0120d891083f939673078ef1a989dae18a

**Федеральное государственное автономное образовательное учреждение высшего образования**

**«Российский университет дружбы народов имени Патриса Лумумбы»**

**Инженерная академия**

(наименование основного учебного подразделения (ОУП) – разработчика ОП ВО)

## **РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ**

### **ИНФОРМАТИКА И ПРОГРАММИРОВАНИЕ**

(наименование дисциплины/модуля)

**Рекомендована МССН для направления подготовки/специальности:**

### **27.03.04 УПРАВЛЕНИЕ В ТЕХНИЧЕСКИХ СИСТЕМАХ**

(код и наименование направления подготовки/специальности)

**Освоение дисциплины ведется в рамках реализации основной профессиональной образовательной программы высшего образования (ОП ВО):**

### **DATA SCIENCE И КОСМИЧЕСКИЕ СИСТЕМЫ**

(наименование (профиль/специализация) ОП ВО)

## 1. ЦЕЛЬ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Дисциплина «Информатика и программирование» входит в программу бакалавриата «Data Science и космические системы» по направлению 27.03.04 «Управление в технических системах» и изучается во 2, 3, 4, 5 семестрах 1, 2, 3 курсов. Дисциплину реализует Кафедра механики и процессов управления. Дисциплина состоит из 29 разделов и 101 тема и направлена на изучение теоретических и практических основ информационных технологий и программирования. Особое внимание уделяется разбору методов решения типовых задач и анализу области их применения в профессиональной деятельности.

Целью освоения дисциплины является формирование фундаментальных знаний и навыков применения технологий программирования для решения широкого спектра задач, необходимых для профессиональной деятельности и освоения последующих дисциплин.

## 2. ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Освоение дисциплины «Информатика и программирование» направлено на формирование у обучающихся следующих компетенций (части компетенций):

Таблица 2.1. Перечень компетенций, формируемых у обучающихся при освоении дисциплины (результаты освоения дисциплины)

Шифр	Компетенция	Индикаторы достижения компетенции (в рамках данной дисциплины)
ОПК-6	Способен разрабатывать и использовать алгоритмы и программы, современные информационные технологии, методы и средства контроля, диагностики и управления, пригодные для практического применения в сфере своей профессиональной деятельности	ОПК-6.1 Знает основные алгоритмы и программы, современные информационные технологии, методы и средства контроля, диагностики и управления, пригодные для практического применения в сфере своей профессиональной деятельности; ОПК-6.2 Умеет применять алгоритмы и программы, современные информационные технологии, методы и средства контроля, диагностики и управления, пригодные для практического применения в сфере своей профессиональной деятельности; ОПК-6.3 Уверенно владеет алгоритмами и и программами, современными информационными технологиями, методами и средствами контроля, диагностикой и управлением, пригодными для практического применения в сфере своей профессиональной деятельности;
ОПК-9	Способен выполнять эксперименты по заданным методикам и обрабатывать результаты с применением современных информационных технологий и технических средств	ОПК-9.1 Знает современные информационные технологии и технические средства; ОПК-9.2 Умеет применять современные информационные технологии и технические средства для обработки результатов экспериментов; ОПК-9.3 Владеет современными информационными технологиями и техническими средствами для выполнения экспериментов и обработки результатов;
ПК-1	Способен собирать, обрабатывать и интерпретировать данные современных научных исследований, необходимые для формирования выводов по соответствующим научным исследованиям, в том числе данные дистанционного зондирования Земли	ПК-1.1 Знает современные методы того, как собирать, обрабатывать и интерпретировать данные современных научных исследований, необходимые для формирования выводов по соответствующим научным исследованиям; ПК-1.2 Умеет применять современные методы и средства для обработки и интерпретации данные научных исследований; ПК-1.3 Владеет основными навыками сбора, обработки и интерпретации данных современных научных исследований, необходимых для формирования выводов по соответствующим научным исследованиям;

## 3. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОП ВО

Дисциплина «Computer Science and Programming» относится к обязательной части блока 1 «Дисциплины (модули)» образовательной программы высшего образования.

В рамках образовательной программы высшего образования обучающиеся также осваивают другие дисциплины и/или практики, способствующие достижению запланированных результатов освоения дисциплины «Computer Science and Programming».

Таблица 3.1. Перечень компонентов ОП ВО, способствующих достижению запланированных результатов освоения дисциплины

Шифр	Наименование компетенции	Предшествующие дисциплины/модули, практики*	Последующие дисциплины/модули, практики*
ОПК-6	Способен разрабатывать и использовать алгоритмы и программы, современные информационные технологии, методы и средства контроля, диагностики и управления, пригодные для практического применения в сфере своей профессиональной деятельности	Introduction to Computing Science;	Automatic Control Theory; Space Flight Mechanics; Research work / Научно-исследовательская работа; Undergraduate Training;
ОПК-9	Способен выполнять эксперименты по заданным методикам и обрабатывать результаты с применением современных информационных технологий и технических средств	Basic Military Training. Life Safety; Introduction to Computing Science;	Undergraduate Training; Technological Training; Analysis of Geoinformation Data; Optimal Control Methods;
ПК-1	Способен собирать, обрабатывать и интерпретировать данные современных научных исследований, необходимые для формирования выводов по соответствующим научным исследованиям, в том числе данные дистанционного зондирования Земли	Introduction to Computing Science;	Virtual and Augmented Reality Technology**; Технологии виртуальной и дополненной реальности**; Optimal Control Methods; Analysis of Geoinformation Data; Space Flight Mechanics; Automatic Control Theory; Research work / Научно-исследовательская работа; Technological Training; Undergraduate Training;

\* - заполняется в соответствии с матрицей компетенций и СУП ОП ВО

\*\* - элективные дисциплины /практики

#### 4. ОБЪЕМ ДИСЦИПЛИНЫ И ВИДЫ УЧЕБНОЙ РАБОТЫ

Общая трудоемкость дисциплины «Информатика и программирование» составляет «17» зачетных единиц.

Таблица 4.1. Виды учебной работы по периодам освоения образовательной программы высшего образования для очной формы обучения.

Вид учебной работы	ВСЕГО, ак.ч.	Семестр(-ы)	Семестр(-ы)	Семестр(-ы)	Семестр(-ы)	
		2	3	4	5	
Контактная работа, ак.ч	263	68	72	51	72	
Лекции (ЛК)	123	34	36	17	36	
Лабораторные работы (ЛР)	140	34	36	34	36	
Практические/семинарские занятия (СЗ)	0	0	0	0	0	
Самостоятельная работа обучающихся, ак.ч.	241	94	81	21	45	
Контроль (экзамен/зачет с оценкой), ак.ч.	108	18	27	36	27	
Общая трудоемкость дисциплины ак.ч.	ак.ч.	612	180	180	108	144
	зач.ед.	17	5	5	3	4

Общая трудоемкость дисциплины «Информатика и программирование» составляет «17» зачетных единиц.

Таблица 4.2. Виды учебной работы по периодам освоения образовательной программы высшего образования для очной формы обучения.

Вид учебной работы	ВСЕГО, ак.ч.	Семестр(-ы)	Семестр(-ы)	Семестр(-ы)	Семестр(-ы)	Семестр(-ы)	
		1	2	3	4	5	
Контактная работа, ак.ч	299	36	68	72	51	72	
Лекции (ЛК)	141	18	34	36	17	36	
Лабораторные работы (ЛР)	158	18	34	36	34	36	
Практические/семинарские занятия (СЗ)	0	0	0	0	0	0	
Самостоятельная работа обучающихся, ак.ч.	313	72	94	81	21	45	
Контроль (экзамен/зачет с оценкой), ак.ч.	108	0	18	27	36	27	
Общая трудоемкость дисциплины ак.ч.	ак.ч.	720	108	180	180	108	144
	зач.ед.	20	3	5	5	3	4

Общая трудоемкость дисциплины «Информатика и программирование» составляет «17» зачетных единиц.

Таблица 4.3. Виды учебной работы по периодам освоения образовательной программы высшего образования для очной формы обучения.

Вид учебной работы	ВСЕГО, ак.ч.	Семестр(-ы)	Семестр(-ы)	Семестр(-ы)	Семестр(-ы)	
		2	3	4	5	
Контактная работа, ак.ч	263	68	72	51	72	
Лекции (ЛК)	123	34	36	17	36	
Лабораторные работы (ЛР)	140	34	36	34	36	
Практические/семинарские занятия (СЗ)	0	0	0	0	0	
Самостоятельная работа обучающихся, ак.ч.	241	94	81	21	45	
Контроль (экзамен/зачет с оценкой), ак.ч.	108	18	27	36	27	
Общая трудоемкость дисциплины ак.ч.	ак.ч.	612	180	180	108	144
	зач.ед.	17	5	5	3	4

## 5. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Таблица 5.1. Содержание дисциплины (модуля) по видам учебной работы\*

Номер раздела	Наименование раздела дисциплины	Наименование темы		Содержание темы	Вид учебной работы*
Раздел 1	Информация и информатика	1.1	Основные понятия. Предмет и задачи информатики	Определение информатики как науки о структуре, свойствах информации, методах её сбора, обработки, хранения, передачи и использования. Основные задачи информатики: автоматизация информационных процессов, создание информационных систем и технологий.	ЛК
		1.2	Информация и её свойства	Понятие информации. Свойства информации: объективность, достоверность, полнота, актуальность, полезность, понятность. Формы представления информации. Меры информации: синтаксическая, семантическая, прагматическая.	ЛК
		1.3	Арифметические и логические основы работы ЭВМ	Системы счисления: позиционные и непозиционные. Двоичная, восьмеричная, шестнадцатеричная системы. Перевод чисел между системами счисления. Логические основы: алгебра логики, базовые логические операции (конъюнкция, дизъюнкция, отрицание). Логические элементы и схемы.	ЛК
		1.4	Кодирование информации	Принципы кодирования информации в компьютере. Кодирование чисел: целые и вещественные типы. Кодирование текстовой информации: таблицы кодировок. Кодирование графической, звуковой и видеоинформации. Понятие сжатия данных.	ЛК
		1.5	Перспективы развития информатики	Современные тенденции развития информатики и вычислительной техники: искусственный интеллект, машинное обучение, большие данные, облачные вычисления, интернет вещей. Направления дальнейшего развития.	ЛК
		1.6	Современные аспекты программирования. Классификация и области применения современных языков программирования	Эволюция языков программирования. Классификация: машинно-ориентированные, процедурные, объектно-ориентированные, функциональные, логические. Области применения: системное программирование, веб-разработка, научные вычисления, мобильная разработка, анализ данных.	ЛК
Раздел 2	Вычислительные техника	2.1	История развития и классификация ЭВМ	Этапы развития вычислительной техники: от механических устройств до современных компьютеров. Поколения ЭВМ. Классификация: суперкомпьютеры, серверы, персональные компьютеры, встроенные системы, мобильные устройства.	ЛК
		2.2	Архитектура ЭВМ. Состав вычислительной системы	Принципы фон Неймана. Основные компоненты: процессор, память, устройства ввода-вывода. Магистрально-модульная архитектура. Понятие о современных архитектурах.	ЛК
		2.3	Принципы функционирования элементов вычислительной системы	Центральный процессор: устройство управления, арифметико-логическое устройство, регистры. Память: оперативная, постоянная, кэш-память. Принципы работы устройств ввода и вывода.	ЛК, ЛР
		2.4	Компьютерные сети. Клиент-серверная архитектура	Классификация сетей: локальные, глобальные, корпоративные. Сетевые протоколы. Модель OSI и стек TCP/IP. Клиент-серверная архитектура: принципы работы, типы серверов. Одноранговые сети.	ЛК, ЛР
Раздел 3	Программное обеспечение	3.1	Системное программное обеспечение	Операционные системы: назначение, функции, классификация. Драйверы устройств. Утилиты. Инструментальное программное обеспечение: компиляторы, интерпретаторы, отладчики, среды разработки.	ЛК, ЛР
		3.2	Прикладное программное	Классификация прикладного программного обеспечения: текстовые процессоры,	ЛК, ЛР

Номер раздела	Наименование раздела дисциплины	Наименование темы		Содержание темы	Вид учебной работы*
			обеспечение	электронные таблицы, системы управления базами данных, графические редакторы, мультимедийные приложения. Системы автоматизации расчётов.	
Раздел 4	Основные понятия моделирования и алгоритмизации	4.1	Этапы решения задачи при помощи ЭВМ	Основные этапы: постановка задачи, формализация, построение алгоритма, программирование, отладка и тестирование, анализ результатов. Важность корректности на каждом этапе.	ЛК
		4.2	Модели и их классификация	Понятие модели. Классификация моделей: материальные и информационные; статические и динамические; детерминированные и стохастические; имитационные. Цели моделирования.	ЛК, ЛР
		4.3	Понятие и свойства алгоритма. Способы описания алгоритма	Определение алгоритма. Свойства: дискретность, определённая, результативность, массовость. Способы описания: словесный, графический (блок-схемы), псевдокод, алгоритмические языки.	ЛК
Раздел 5	Язык программирования Python	5.1	Интерпретатор. Базовый синтаксис. Модель памяти. Типы данных	Установка и запуск интерпретатора Python. Базовый синтаксис: отступы, комментарии, операторы. Модель памяти: ссылочная модель, объекты и переменные. Основные типы данных: числа, строки, списки, кортежи, словари, множества.	ЛК, ЛР
		5.2	Логические конструкции. Циклы и ветвления	Условный оператор: логические выражения, вложенные условия. Циклы: цикл со счётчиком, цикл с условием. Операторы управления циклом. Генераторы списков.	ЛК, ЛР
		5.3	Функции. Передача аргументов. Область видимости. Стек вызовов	Определение и вызов функций. Параметры и аргументы: позиционные, именованные, значения по умолчанию. Передача аргументов по ссылке. Области видимости переменных: локальная, глобальная, нелокальная. Стек вызовов и рекурсия.	ЛК, ЛР
		5.4	Работа с файлами. Свойства и виды файлов. Сериализация данных	Открытие, чтение и запись файлов. Текстовые и бинарные файлы. Менеджер контекста. Сериализация данных: преобразование объектов в формат хранения или передачи. Модули для сериализации.	ЛК, ЛР
		5.5	Блочная организация программы. Модули и пакеты. Менеджер пакетов pip	Разделение программы на модули. Импорт модулей. Пакеты как иерархические структуры модулей. Установка сторонних библиотек с помощью менеджера пакетов pip. Управление зависимостями.	ЛК, ЛР
Раздел 6	Библиотеки Python для решения научных и прикладных задач	6.1	Визуализация данных при помощи библиотеки Matplotlib	Построение графиков: линейные, точечные, столбчатые диаграммы. Настройка осей, подписей, легенды. Создание нескольких графиков на одном поле. Сохранение изображений.	ЛК, ЛР
		6.2	Решение задач статистики и линейной алгебры при помощи библиотек NumPy и Pandas	Библиотека NumPy: многомерные массивы, операции над массивами, математические функции. Библиотека Pandas: структуры данных (Series, DataFrame), загрузка и обработка данных, группировка, агрегация, работа с пропущенными значениями.	ЛК, ЛР
Раздел 7	Парадигмы программирования	7.1	Основные парадигмы и их особенности: процедурное программирование, объектно-ориентированное программирование, функциональное программирование	Процурное программирование: программы как последовательность инструкций, процедуры и функции. Объектно-ориентированное программирование: объекты, классы, основные принципы. Функциональное программирование: функции как объекты первого класса, отсутствие побочных эффектов.	ЛК, ЛР
		7.2	Объектно-ориентированное программирование в языке Python. Инкапсуляция,	Определение классов и создание объектов. Атрибуты и методы. Инкапсуляция: приватные и защищённые атрибуты. Наследование: базовые и производные классы, переопределение методов. Полиморфизм: единый интерфейс для разных типов.	ЛК, ЛР

Номер раздела	Наименование раздела дисциплины	Наименование темы		Содержание темы	Вид учебной работы*
			полиморфизм, наследование. Классы и объекты. Наследование классов		
		7.3	Функциональное программирование в языке Python. Анонимные функции: синтаксис и контекст использования. Декораторы функций	Функции высшего порядка. Анонимные функции: синтаксис и применение с функциями сортировки и фильтрации. Декораторы функций: обёртка функций, добавление функциональности без изменения исходного кода.	ЛК
		7.4	Визуально-блочное программирование, как инструмент создания и управления VR-мирами	Понятие визуально-блочного программирования. Принципы перетаскивания блоков. Применение в образовательных средах и для создания виртуальных миров. Примеры сред.	ЛК, ЛР
Раздел 8	Структуры данных	8.1	Базовые структуры данных и их свойства	Основные структуры: массив, список, стек, очередь, связный список, дерево, граф. Свойства: упорядоченность, уникальность элементов, способы доступа и вставки.	ЛК, ЛР
		8.2	Стандартные структуры данных языка Python и особенности работы с ними	Списки: изменяемость, методы. Кортежи: неизменяемость. Словари: пары ключ-значение, хеширование. Множества: уникальность элементов, операции над множествами.	ЛК, ЛР
		8.3	Графовая структура данных. Библиотеки Python реализующие графовую структуру данных и особенности работы с ними	Понятие графа: вершины и рёбра, ориентированные и неориентированные графы. Представление графов: списки смежности, матрицы смежности. Библиотеки для работы с графами, основные алгоритмы на графах.	ЛК, ЛР
Раздел 9	Алгоритмы	9.1	Понятие вычисления и вычислимости. Классификация алгоритмов. Машины Тьюринга.	Понятие вычислимого процесса. Проблема вычислимости. Машина Тьюринга: формальная модель алгоритма, лента, управляющее устройство. Классификация алгоритмов по различным признакам.	ЛК
		9.2	Оценка сложности алгоритмов	Временная и пространственная сложность. Асимптотический анализ. Обозначения: нотации для оценки верхней, нижней и точной границ. Сравнение эффективности алгоритмов.	ЛК
		9.3	Алгоритмы сортировки	Основные алгоритмы сортировки: пузырьковая сортировка, сортировка вставками, сортировка выбором, быстрая сортировка, сортировка слиянием. Сравнение их эффективности.	ЛК, ЛР
		9.4	Алгоритмы поиска	Линейный поиск. Двоичный (бинарный) поиск в упорядоченном массиве. Поиск подстроки в строке. Хеширование как метод быстрого поиска.	ЛК, ЛР
		9.5	Алгоритмы на графах	Обходы графа: поиск в глубину и поиск в ширину. Поиск кратчайших путей: алгоритмы для графов без отрицательных рёбер. Понятие о минимальных остовных деревьях.	ЛК, ЛР
Раздел 10	Библиотеки Python для решения научных и	10.1	Функциональные возможности библиотеки	SciPy как расширение NumPy. Подмодули для оптимизации, интегрирования, интерполяции, линейной алгебры, обработки сигналов и изображений. Решение	ЛК, ЛР

Номер раздела	Наименование раздела дисциплины	Наименование темы		Содержание темы	Вид учебной работы*
	прикладных задач		SciPy и особенности работы с ними	дифференциальных уравнений.	
		10.2	Функциональные возможности библиотеки SymPy и особенности работы с ними	SymPy как библиотека для символьных вычислений. Определение символов, упрощение выражений, дифференцирование, интегрирование, решение уравнений в символьном виде.	ЛК, ЛР
Раздел 11	Основы операционных систем	11.1	История развития и основные функции операционных систем	Этапы развития операционных систем. Основные функции: управление процессами, памятью, вводом-выводом, файловой системой, обеспечение пользовательского интерфейса.	ЛК
		11.2	Основы работы в командном интерпретаторе	Командная строка: приглашение, команды. Навигация по файловой системе, создание и удаление файлов и каталогов, запуск программ. Перенаправление ввода-вывода, конвейеры.	ЛК, ЛР
		11.3	Архитектурные особенности построения операционных систем	Монолитное ядро, микроядро, гибридное ядро. Модульность. Основные компоненты ядра. Режимы работы процессора: пользовательский и привилегированный.	ЛК, ЛР
		11.4	Управление процессами и памятью	Процессы и потоки: состояния, контекст, планирование. Алгоритмы планирования. Управление памятью: виртуальная память, страничная и сегментная организация, подкачка.	ЛК, ЛР
		11.5	Управление вводом-выводом	Драйверы устройств. Буферизация, кэширование. Обработка прерываний. Файловые системы: структура, типы, основные операции.	ЛК, ЛР
Раздел 12	Системы контроля версий (СКВ)	12.1	История развития СКВ. Основные концепции и термины. Классификация и современные СКВ	Эволюция систем контроля версий: локальные, централизованные, распределённые. Основные понятия: репозиторий, коммит, ветка, тег. Современные системы.	ЛК, ЛР
		12.2	Использование Git и организация рабочего процесса создания программного обеспечения	Основные команды Git: инициализация репозитория, добавление изменений, фиксация, просмотр истории. Работа с ветками: создание, слияние, разрешение конфликтов. Удалённые репозитории. Организация совместной работы.	ЛК, ЛР
Раздел 13	Основы языка программирования C	13.1	История развития, особенности и область применения языка C	Создание языка C. Особенности: эффективность, низкоуровневые возможности, переносимость. Области применения: системное программирование, встроенные системы, операционные системы.	ЛК
		13.2	Объявление и определение переменных. Типы переменных. Преобразование типов.	Базовые типы данных: целые, вещественные, символьные. Объявление и определение переменных. Инициализация. Явное и неявное преобразование типов.	ЛК, ЛР
		13.3	Арифметические и логические операторы. Побитовые операторы. Приоритет и порядок вычисления.	Арифметические операции. Логические операторы. Побитовые операторы: работа с отдельными битами. Приоритет операций. Порядок вычисления выражений.	ЛК, ЛР
		13.4	Управляющие конструкции. Ветвление и циклы, операторы безусловного	Условный оператор. Тернарный оператор. Оператор множественного выбора. Циклы с предусловием, с постусловием, со счётчиком. Операторы безусловного перехода.	ЛК, ЛР

Номер раздела	Наименование раздела дисциплины	Наименование темы		Содержание темы	Вид учебной работы*
			перехода и множественного выбора		
Раздел 14	Функции и структура программы	14.1	Функции. Синтаксические конструкции для работы с функциями: объявление, определение, вызов. Рекурсия. Стек вызовов. Блочная структура программы	Объявление (прототип) и определение функции. Передача аргументов: по значению. Возврат значения. Рекурсивные функции. Стек вызовов. Блочная структура: локальные блоки.	ЛК
		14.2	Внешние переменные и область видимости. Статические и регистровые переменные. Заголовочные файлы.	Глобальные и локальные переменные. Область видимости и время жизни. Статические переменные: сохранение значения между вызовами. Регистровые переменные. Заголовочные файлы: объявление функций и макросов.	ЛК, ЛР
		14.3	Процесс компиляции программ. Препроцессор, включение файла, макроподстановка, условная компиляция	Этапы: препроцессирование, компиляция, ассемблирование, компоновка. Директивы препроцессора: включение файлов, определение макросов, условная компиляция.	ЛР
Раздел 15	Указатели и массивы	15.1	Указатели и адреса. Указатели и аргументы функции	Понятие указателя как переменной, хранящей адрес. Оператор взятия адреса и разыменования. Передача аргументов по указателю для изменения значений внутри функции.	ЛК, ЛР
		15.2	Массивы. Адресная арифметика	Определение и инициализация массивов. Связь массивов и указателей. Адресная арифметика: сложение и вычитание указателей. Доступ к элементам массива через указатель.	ЛК
		15.3	Указатели на указатели. Многомерные массивы	Указатель на указатель. Представление многомерных массивов в памяти. Доступ к элементам многомерных массивов через указатели.	ЛК, ЛР
		15.4	Аргументы командной строки. Указатели на функции. Сложные объявления	Передача аргументов в функцию main. Указатели на функции: объявление, присваивание, вызов. Правила чтения сложных объявлений с указателями и массивами.	ЛК, ЛР
Раздел 16	Структуры	16.1	Основы работы со структурами. Структуры и функции. Указатели на структуры	Определение структурного типа. Объявление переменных-структур. Доступ к полям. Передача структур в функции по значению и по указателю.	ЛК, ЛР
		16.2	Определение новых типов	Использование ключевого слова для создания псевдонимов типов. Улучшение читаемости кода.	ЛК
		16.3	Объединения и битовые поля	Объединения: хранение разных типов в одной области памяти. Битовые поля: экономия памяти при хранении флагов.	ЛК, ЛР
Раздел 17	Операции ввода-вывода	17.1	Стандартные средства ввода-вывода	Функции форматированного и неформатированного ввода-вывода. Стандартные потоки: ввод, вывод, вывод ошибок.	ЛК, ЛР
		17.2	Списки аргументов переменной длины.	Создание функций с переменным числом аргументов. Макросы для работы с такими списками. Форматированный ввод.	ЛК, ЛР

Номер раздела	Наименование раздела дисциплины	Наименование темы		Содержание темы	Вид учебной работы*
			Форматированный ввод		
		17.3	Чтение и запись файлов	Открытие и закрытие файлов. Режимы доступа. Чтение и запись символов, строк, блоков данных. Позиционирование в файле.	ЛК, ЛР
		17.4	Обработка ошибок	Проверка ошибок при выполнении операций ввода-вывода. Коды ошибок.	ЛК, ЛР
Раздел 18	Стандартная библиотека	18.1	Операции со строками. Анализ, классификация и преобразование символов	Функции для работы со строками: копирование, конкатенация, сравнение, поиск. Функции классификации символов: проверка на букву, цифру и т.д. Преобразование регистра.	ЛК, ЛР
		18.2	Выполнение команд. Управление памятью	Выполнение системных команд из программы. Функции динамического выделения и освобождения памяти. Управление динамической памятью.	ЛК
		18.3	Математические функции. Генератор случайных чисел	Математические функции: тригонометрические, экспоненциальные, логарифмические. Генерация псевдослучайных чисел.	ЛК, ЛР
Раздел 19	Основы языка программирования C++	19.1	История развития, особенности и область применения языка C. Отличия языков C и C++	Создание языка C++. Поддержка объектно-ориентированного программирования. Отличия от C: ввод-вывод, ссылки, пространства имён, перегрузка функций.	ЛК
		19.2	Типы и объявления. Пространства имен. Указатели, ссылки, массивы и структуры	Базовые типы данных. Объявление переменных. Пространства имён для избегания конфликтов имён. Ссылки как альтернатива указателям. Массивы и структуры в C++.	ЛК, ЛР
		19.3	Выражения и операторы. Функции	Операторы языка C++. Управляющие конструкции. Функции: перегрузка функций, значения по умолчанию, передача по ссылке.	ЛК, ЛР
		19.4	Исключения. Ключевые слова throw, catch	Механизм обработки исключений: блоки для отслеживания ошибок, генерация исключений, перехват исключений.	ЛК, ЛР
		19.5	Исходные файлы и программы. Раздельная компиляция	Разделение программы на несколько исходных файлов. Заголовочные файлы. Процесс раздельной компиляции и компоновки.	ЛК, ЛР
Раздел 20	Механизмы абстракции (ООП)	20.1	Классы и объекты. Члены класса. Конструкторы и деструкторы. Композиция классов. Модификаторы доступа. Перегрузка методов класса.	Определение класса. Создание объектов. Поля и методы. Конструкторы: по умолчанию, с параметрами, копирования. Деструктор. Вложение объектов (композиция). Модификаторы доступа. Перегрузка методов.	ЛК, ЛР
		20.2	Перегрузка операций. Функции-операции. Операции приведения типов. Друзья класса	Перегрузка операторов как функций-членов или внешних функций. Перегрузка операторов приведения типа. Дружественные функции и классы для доступа к закрытым членам.	ЛК, ЛР
		20.3	Наследование классов. Производные классы. Виртуальные функции. Классовые иерархии и абстрактные классы	Наследование: базовый и производный классы. Порядок вызова конструкторов и деструкторов. Виртуальные функции для позднего связывания. Полиморфизм. Абстрактные классы.	ЛК, ЛР
		20.4	Шаблоны. Определение шаблона. Конкретизация	Понятие шаблонов для создания обобщённого кода. Шаблоны функций. Шаблоны классов. Конкретизация шаблонов конкретными типами. Частичная и полная	ЛК, ЛР

Номер раздела	Наименование раздела дисциплины	Наименование темы		Содержание темы	Вид учебной работы*
			шаблонов. Проверка типов. Шаблоны функции. Специализация	специализация.	
Раздел 21	Обработка исключений	21.1	Обработка ошибок. Группировка исключений	Традиционные способы обработки ошибок и их недостатки. Исключения как альтернатива. Иерархия классов исключений.	ЛК, ЛР
		21.2	Перехват исключений. Управление ресурсами	Блоки перехвата. Порядок перехвата. Повторная генерация исключения. Идиома захвата ресурса при инициализации для безопасного управления ресурсами.	ЛК, ЛР
		21.3	Спецификация исключений	Указание того, какие исключения может генерировать функция. Спецификации.	ЛК, ЛР
		21.4	Исключения и эффективность. Альтернативы обработке ошибок	Влияние исключений на производительность. Сравнение исключений с другими механизмами обработки ошибок.	ЛК, ЛР
Раздел 22	Иерархии классов	22.1	Проектирование иерархии классов. Традиционные иерархии классов	Принципы проектирования иерархий. Абстрактные базовые классы. Примеры классических иерархий.	ЛК
		22.2	Множественное наследование и контроль доступа	Наследование от нескольких базовых классов. Проблема ромбовидного наследования. Виртуальное наследование. Управление доступом при наследовании.	ЛК
Раздел 23	Стандартная библиотека STL	23.1	Стандартные контейнеры	Последовательные контейнеры: вектор, список, дек. Ассоциативные контейнеры: множество, отображение, мультимножество, мультиотображение. Неупорядоченные контейнеры.	ЛК, ЛР
		23.2	Алгоритмы и классы функциональных объектов	Обобщённые алгоритмы: сортировка, поиск, копирование, преобразование. Функциональные объекты (функторы). Лямбда-выражения.	ЛК, ЛР
		23.3	Итераторы и аллокаторы	Итераторы как обобщение указателей. Категории итераторов. Аллокаторы для управления памятью в контейнерах.	ЛК, ЛР
		23.4	Строки и потоки	Класс для работы со строками. Поточный ввод-вывод: строковые потоки.	ЛК, ЛР
		23.5	Классы для математических вычислений	Комплексные числа. Пары. Функции для работы с последовательностями.	ЛК, ЛР
Раздел 24	Технология программирования	24.1	Основные понятия и подходы	Программное обеспечение как продукт. Технология программирования: цели, принципы. Инженерия программного обеспечения.	ЛК
		24.2	Проблемы разработки сложных программных систем	Сложность, согласованность, изменяемость, невидимость. Проблемы управления сроками, бюджетом и качеством.	ЛК
		24.3	Блочный-иерархический подход к созданию сложных систем	Принцип декомпозиции. Разделение на подсистемы, модули, компоненты. Иерархическая структура.	ЛК
		24.4	Жизненный цикл и этапы разработки	Основные этапы: анализ требований, проектирование, реализация, тестирование, внедрение, сопровождение. Модели жизненного цикла.	ЛК
		24.5	Оценка качества процессов создания программного обеспечения	Стандарты и метрики качества. Производительность, надёжность, сопровождаемость.	ЛК

Номер раздела	Наименование раздела дисциплины	Наименование темы		Содержание темы	Вид учебной работы*
Раздел 25	Приемы обеспечения технологичности программных продуктов	25.1	Технологичность программного обеспечения. Модули и их свойства	Понятие технологичности. Модули: связность и сцепление. Хорошие свойства модулей.	ЛК
		25.2	Нисходящая и восходящая разработка	Стратегии разработки: от общего к частному (нисходящая) и от частного к общему (восходящая). Их сочетание.	ЛК
		25.3	Структурное и «неструктурное» программирование. Средства описания структурных алгоритмов	Принципы структурного программирования: запрет безусловных переходов. Средства описания: блок-схемы, Nassi-Shneiderman диаграммы.	ЛК, ЛР
		25.4	Стиль оформления программы. Эффективность и технологичность	Форматирование кода, именование переменных, комментирование. Влияние стиля на сопровождаемость. Компромисс между эффективностью и технологичностью.	ЛК, ЛР
Раздел 26	Определение требований к программному обеспечению	26.1	Классификация программных продуктов по функциональному признаку. Основные эксплуатационные требования	Классификация: системное, прикладное, инструментальное ПО. Требования: функциональные, нефункциональные (надёжность, производительность, безопасность).	ЛК
		26.2	Разработка технического задания. Принципиальные решения начальных этапов проектирования	Состав и структура технического задания. Выбор архитектуры, платформы, технологий.	ЛК
Раздел 27	Структурный подход	27.1	Анализ требований и определение спецификации при структурном подходе. Диаграммы переходов состояний, функциональные диаграммы, диаграммы потоков данных. Структуры данных и диаграммы отношений компонентов данных. Математические модели задач	Методы структурного анализа. Диаграммы переходов состояний. Функциональные диаграммы. Диаграммы потоков данных. Моделирование структур данных.	ЛК, ЛР
		27.2	Проектирование программного обеспечения при структурном подходе. Структурная и функциональная схемы. Пошаговая детализация. Карты Константайна. Проектирования структур	Построение структурных и функциональных схем. Метод пошаговой детализации. Карты Константайна для визуализации структуры. Декомпозиция данных как основа проектирования.	ЛК

Номер раздела	Наименование раздела дисциплины	Наименование темы	Содержание темы	Вид учебной работы*	
			данных. Проектирование на основе декомпозиции данных. Case-технологии		
Раздел 28	Объектный подход	28.1	Анализ требований и определение спецификации при объектном подходе. UML. Определение вариантов использования. Построение концептуальной модели предметной области. Описание поведения	Унифицированный язык моделирования. Диаграммы вариантов использования. Концептуальные модели. Диаграммы деятельности и последовательностей для описания поведения.	ЛК, ЛР
		28.2	Проектирование программного обеспечения при объектном подходе. Разработка структуры. Определение отношений между объектами и классами. Проектирование классов. Компоновка. Размещение распределенных программных систем. Спиральная модель разработки	Диаграммы классов. Отношения: ассоциация, агрегация, композиция, наследование. Проектирование классов и компонентов. Диаграммы развёртывания. Спиральная модель жизненного цикла.	ЛК
Раздел 29	Тестирование программных продуктов	29.1	Виды контроля качества. Ручной контроль. Структурное и функциональное тестирование	Методы обеспечения качества: обзоры кода, инспекции. Ручное тестирование. Структурное тестирование (белого ящика). Функциональное тестирование (чёрного ящика).	ЛК, ЛР
		29.2	Модульное, комплексное и оценочное тестирование	Тестирование отдельных модулей. Интеграционное тестирование взаимодействия модулей. Системное тестирование. Приёмочное тестирование. Оценочное тестирование характеристик.	ЛК, ЛР

\* - заполняется только по ОЧНОЙ форме обучения: ЛК – лекции; ЛР – лабораторные работы; СЗ – практические/семинарские занятия.

## 6. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Таблица 6.1. Материально-техническое обеспечение дисциплины

Тип аудитории	Оснащение аудитории	Специализированное учебное/лабораторное оборудование, ПО и материалы для освоения дисциплины (при необходимости)
Лекционная	Аудитория для проведения занятий лекционного типа, оснащенная комплектом специализированной мебели; доской (экраном) и техническими средствами мультимедиа презентаций.	
Компьютерный класс	Компьютерный класс для проведения занятий, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, оснащенная персональными компьютерами (в количестве 14 шт.), доской (экраном) и техническими средствами мультимедиа презентаций.	
Семинарская	Аудитория для проведения занятий семинарского типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, оснащенная комплектом специализированной мебели и техническими средствами мультимедиа презентаций.	
Для самостоятельной работы	Аудитория для самостоятельной работы обучающихся (может использоваться для проведения семинарских занятий и консультаций), оснащенная комплектом специализированной мебели и компьютерами с доступом в ЭИОС.	

\* - аудитория для самостоятельной работы обучающихся указывается **ОБЯЗАТЕЛЬНО!**

## 7. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

### Основная литература:

1. Информатика. Базовый курс. Симонович С.В., СПб.: Питер, 2011 — 640 с.
2. Изучаем Python. Том 1. 5-е издание. М. Лутц, СПб.: Диалектика, 2019 — 832 с.
3. Python 3. Самое необходимое. Прохоренок Н., Дронов В., СПб.: БХВ-Петербург, 2019 — 610 с.
4. Язык программирования С. Брайан У. Керниган, Д.М. Ритчи, М.: Вильямс, 2019 — 288 с.
5. Как программировать на С. 7-е издание. Х. Дейтел, П. Дейтел, М.: БИНОМ, 2017 — 1000 с.
6. Язык программирования С. Лекции и упражнения. Стивен Прата. М.: Вильямс, 2015 — 928 с.
7. Алгоритмы. Справочник с примерами на С, С++, Java и Python. Хайнеман Дж., Поллис Г., Селков С., СПб.: ООО "Альфа-книга", 2017 — 432 с.
8. Язык программирования С++. Страуструп Б., Мартынов Н.Н., М: Бином, 2011. - 1135 с.
9. Как программировать на С++. 8-е издание. Х. Дейтел, П. Дейтел, М.: Бином, 2020 — 1032 с.
10. С++. Священные знания. Дьюхерст С., СПб.: Символ Плюс, 2012 – 240 с.
11. Паттерны объектно-ориентированного проектирования. Гамма Э., Хелм Р., Джонсон Р., Влиссидес Дж., СПб.: Питер, 2020 — 448 с.
12. Алгоритмы. Справочник с примерами на С, С++, Java и Python. Хайнеман Дж., Поллис Г., Селков С., СПб.: ООО "Альфа-книга", 2017 — 432 с.

### Дополнительная литература:

1. Практическое программирование на Python
  - The Computer Science Book: A complete introduction to computer science in one book. Johnson Thomas, Canada: Leanpub, 2020, — 410 p.
  - Автоматизация рутинных задач с помощью Python: практическое руководство для начинающих. Свейгарт Эл., М.: "ИД Вильямс", 2017 — 592 с.
  - Классические задачи Computer Science на языке Python. Копец Д. СПб.: Питер, 2020 — 256 с.

- Большая книга проектов Python. Свейгарт Эл. СПб.: Питер, 2022 — 432 с.;
- Изучаем Python: программирование игр, визуализация данных, веб-приложения. Мэттиз Э. СПб.: Питер, 2020 — 512 с.

## 2. Алгоритмы, инструменты и теория

- Алгоритмы: построение, анализ и реализация на языке программирования Си. Ворожцов А.В., Винокуров Н.А., Москва: МФТИ, 2007 — 452 с.
- Программирование и информатика. Антонюк В.А., Иванов А.П., Москва: Физический фак. МГУ им. М. В. Ломоносова, 2015 — 64 с.
- Pro Git. Version 2.1.x. Scott Chacon, Ben Straub, USA, New York: Apress, 2020 — 506 p. URL: <https://git-scm.com/book/en/v2>

## 3. Объектно-ориентированное программирование

- Объектно-ориентированное мышление. Вайсфельд М., СПб.: Питер, 2014 — 304 с.
- Объектно-ориентированное программирование: Практикум. Павловская Т.А., Щупак Ю.А., СПб.: Питер, 2006. — 265 с.
- Структуры и алгоритмы обработки данных: объектно-ориентированный подход и реализация на C++. Кубенский А.А. СПб.:БХВ-Петербург, 2004 — 464 с.

### *Ресурсы информационно-телекоммуникационной сети «Интернет»:*

#### 1. ЭБС РУДН и сторонние ЭБС, к которым студенты университета имеют доступ на основании заключенных договоров

- Электронно-библиотечная система РУДН – ЭБС РУДН <http://lib.rudn.ru/MegaPro/Web>
- ЭБС «Университетская библиотека онлайн» <http://www.biblioclub.ru>
- ЭБС Юрайт <http://www.biblio-online.ru>
- ЭБС «Консультант студента» [www.studentlibrary.ru](http://www.studentlibrary.ru)
- ЭБС «Троицкий мост»

#### 2. Базы данных и поисковые системы

- электронный фонд правовой и нормативно-технической документации <http://docs.cntd.ru/>
- поисковая система Яндекс <https://www.yandex.ru/>
- поисковая система Google <https://www.google.ru/>
- реферативная база данных SCOPUS <http://www.elsevierscience.ru/products/scopus/>

### *Учебно-методические материалы для самостоятельной работы обучающихся при освоении дисциплины/модуля\*:*

#### 1. Курс лекций по дисциплине «Computer Science and Programming».

\* - все учебно-методические материалы для самостоятельной работы обучающихся размещаются в соответствии с действующим порядком на странице дисциплины **в ТУИС!**

**РАЗРАБОТЧИКИ**

Доцент

Должность

**РУКОВОДИТЕЛЬ ОП ВО**

Профессор

Должность

**РУКОВОДИТЕЛЬ БУП**

Заведующий кафедрой

Должность

Салтыкова О.А.

Фамилия И.О

Разумный Ю.Н.

Фамилия И.О

Разумный Ю.Н.

Фамилия И.О