

Министерство науки и высшего образования Российской Федерации  
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ (НИ ТГУ)

Институт прикладной математики и компьютерных наук

УТВЕРЖДАЮ:

Директор

  
А. В. Замятин

« 14 » июня 20 23 г.

Рабочая программа дисциплины

**Основы программирования**

по направлению подготовки

**02.03.03 Математическое обеспечение и администрирование информационных систем**

**Направленность (профиль) подготовки :**

**DevOps-инженерия в администрировании инфраструктуры ИТ-разработки**

Форма обучения

**Очная**

Квалификация

**Бакалавр**

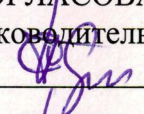
Год приема

**2023**

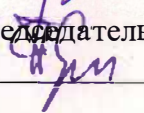
Код дисциплины в учебном плане: Б1.О.04.01

СОГЛАСОВАНО:

Руководитель ОП

  
С.П. Сущенко

Председатель УМК

  
С.П. Сущенко

Томск – 2023

## **1. Цель и планируемые результаты освоения дисциплины**

Целью освоения дисциплины является формирование следующих компетенций:

– ОПК-6 – способность понимать принципы работы современных информационных технологий и использовать их для решения задач профессиональной деятельности;

– ОПК-2 – способность применять компьютерные/суперкомпьютерные методы, современное программное обеспечение, в том числе отечественного происхождения, для решения задач профессиональной;

- ОПК-3. - Способен понимать и применять современные информационные технологии, в том числе отечественные, при создании программных продуктов и программных комплексов различного назначения

– ОПК-5 – устанавливать и сопровождать программное обеспечение информационных систем и баз данных, в том числе отечественного происхождения, с учетом информационной безопасности

Результатами освоения дисциплины являются следующие индикаторы достижения компетенций:

ИОПК-6.1 Обладает необходимыми знаниями в области информационных технологий, в том числе понимает принципы их работы.

ИОПК-6.2 Применяет знания, полученные в области информационных технологий, при решении задач профессиональной деятельности.

ИОПК-6.3 Применяет знания, полученные в области информационных технологий, при решении задач профессиональной деятельности.

ИОПК-3.3 Использует современные информационные технологии, в том числе отечественного производства на всех этапах разработки программных систем

ИОПК-2.3 Использует инструментальные средства высокопроизводительных вычислений в научной и практической деятельности.

ИОПК-2.2 Использует методы высокопроизводительных вычислительных технологий, современного программного обеспечения, в том числе отечественного происхождения.

ИОПК-2.1 Обладает необходимыми знаниями основных концепций современных вычислительных систем.

ИОПК-5.3 Использует необходимые знания для сопровождения программного обеспечения информационных систем и баз данных, в том числе отечественного происхождения, с учетом информационной безопасности.

ИОПК-5.2 Устанавливает программное и аппаратное обеспечение.

ИОПК-5.1 Определяет порядок и особенности процесса установки программного и аппаратного обеспечения для информационных и автоматизированных систем.

## **2. Задачи освоения дисциплины**

– изучение принципов разработки алгоритмических и программных решений в области системного и прикладного программирования;

– формирование навыков оценки эффективности разрабатываемых алгоритмов;

– изучение основ языков программирования Паскаль и С++ для составления программ умеренной сложности.

Дисциплина относится к обязательной части образовательной программы.

Дисциплина входит в модуль «Разработка программного обеспечения».

## **4. Семестр(ы) освоения и форма(ы) промежуточной аттестации по дисциплине**

Первый семестр, экзамен

Второй семестр, экзамен

## **5. Входные требования для освоения дисциплины**

Для успешного освоения дисциплины требуются компетенции, сформированные в ходе освоения образовательных программ предшествующего уровня образования.

## **6. Язык реализации**

Русский

## **7. Объем дисциплины**

Общая трудоемкость дисциплины составляет 13 з.е., 468 часов, из которых:

-лекции: 64 ч.

-лабораторные: 144 ч.

в том числе практическая подготовка: 144 ч.

Объем самостоятельной работы студента определен учебным планом.

## **8. Содержание дисциплины, структурированное по темам**

### **Тема 1. Основы программирования на языке Паскаль**

Трансляция и выполнение программы. Типы данных. Операторы. Процедуры и функции.

### **Тема 2. Тестирование и отладка программ**

Задачи тестирования. Создание тестов. Генерация тестов большой размерности.

### **Тема 3. Доказательство свойств программ**

Вход и выход алгоритма. Пред- и постусловие алгоритма. Методы доказательства правильности. Трудоемкость алгоритмов, ее вычисление.

### **Тема 4. Основные алгоритмы и их трудоемкость**

Вычисление рекуррентных последовательностей, сумм, произведений. Примеры рекуррентных алгоритмов, их трудоемкость.

### **Тема 5. Простые алгоритмы сортировки и поиска и их трудоемкость**

Простые алгоритмы упорядочения элементов в массиве. Косвенная упорядоченность. Поиск элементов в массиве. Алгоритм слияния упорядоченных массивов.

### **Тема 6. Простые рекурсивные алгоритмы**

Простые рекурсивные функции. Локальные и глобальные переменные. Выполнение рекурсивного алгоритма с помощью стека. Глубина рекурсии. Трудоемкость рекурсивных алгоритмов.

### **Тема 7. Файлы в Паскале. Взаимодействие с операционной системой**

Файловая система, действия над файлами. Стандартные файлы ввода и вывода в Паскале. Выполняемая программа с параметрами. Обработка параметров. Программа из модулей, трансляция программы по частям.

### **Тема 8. Списочные структуры**

Тип указатель. Списочные структуры. Создание и удаление элементов списка. Реализация стека и очереди с помощью списка. Рекурсивные алгоритмы со списками.

### **Тема 9. Рекурсивные алгоритмы бэктрекинга**

Рекурсивный алгоритм генерации перестановок чисел, его трудоемкость. Алгоритмы генерации размещений, сочетаний. Алгоритмы, реализующие бэктрекинг с отсечением.

#### **Тема 10. Алгоритмы над множествами**

Представление множества в виде логического (битового) массива. Множество в виде массива неупорядоченных и упорядоченных номеров. Алгоритмы над множествами, использующие идею слияния. Множество в виде списка неупорядоченных и упорядоченных номеров.

#### **Тема 11. Алгоритмы со строками и таблицами**

Перекодировка строк. Задача поиска подстроки символов в строке. Использование косвенной сортировки для быстрого поиска в таблице. Объединение таблиц, имеющих одинаковые столбцы. Примеры алгоритмов с таблицами.

#### **Тема 12. Основы программирования на языке Си**

Трансляция и выполнение программы на языке Си. Структура программы. Стандартные средства ввода и вывода. Типы данных. Указатели и массивы. Функции, их вызов и описание.

#### **Тема 13. Простые программы на Си**

Программы, содержащие выделение памяти для массивов. Функция сортировки слиянием массивов. Списки в Си, их создание и удаление. Функция сравнения строк символов с перекодировкой.

#### **Тема 14. Синтаксис и семантика языка программирования**

Символы и слова (лексемы) языка программирования. Грамматические конструкции (понятия) в языке. Порождающая грамматика в виде правил Бэкуса-Наура для задания синтаксиса языка. Семантика языка. Примеры правил Бэкуса-Наура для языка Си.

#### **Тема 15. Алгоритмы с векторами и матрицами**

Действия с векторами и матрицами: сложение, умножение, скалярное произведение. Вычисление определителя приведением матрицы к диагональному виду. Определение ранга матрицы. Решение системы линейных уравнений методом Гаусса с выбором главного элемента. Обращение матрицы методом Гаусса.

#### **Тема 16. Простые алгоритмы над графами**

Представление графа матрицей смежности или матрицей расстояний. Представление графа массивом списков ребер. Представление графа списочными структурами. Алгоритм просмотра графа вширь для обоих способов представления, их трудоемкость. Задача нахождения кратчайшего маршрута в лабиринте. Рекурсивный алгоритм просмотра графа вглубь и его трудоемкость. Задача выделения компонент связности. Алгоритм топологического упорядочения вершин графа и его трудоемкость.

#### **Тема 17. Циклы и пути в графах**

Эйлеровы циклы и пути в ориентированном и неориентированном графах. Гамильтоновы циклы и пути в ориентированном и неориентированном графах, Кратчайшие пути во взвешенном графе, алгоритм Дейкстры.

#### **Тема 18. Разработка больших программ**

Программный продукт. Программная документация. Этапы разработки программного продукта. Проект системы. Проектирование и разработка сверху-вниз и

снизу-вверх. Использование заглушек. Одновременное проектирование, разработка и тестирование. Тестирующие программы. Независимое тестирование. Тестирование документации. Организационные проблемы создания больших программных систем.

### 9. Текущий контроль по дисциплине

Текущий контроль по дисциплине основан на применении 200-балльной шкалы оценивания в каждом семестре. Проводится оценивание выполнения контрольных работ (по 100-балльной шкале) и лабораторных заданий (по 100-балльной шкале). Критерии оценивания публикуются в методических материалах к дисциплине. Результаты текущего контроля определяются по общей сумме баллов и фиксируются в форме контрольной точки не менее одного раза в семестр. Обучающиеся, набравшие не менее 35 баллов, выполнившие не менее одной контрольной работы и не менее одного обязательного задания, получают аттестацию. Обучающиеся, не выполнившие хотя бы одно из перечисленных выше требований, считаются не аттестованными.

Практическая подготовка оценивается по результатам выполненных лабораторных работ.

### 10. Порядок проведения и критерии оценивания промежуточной аттестации

Теоретические и практические результаты формируются компетенциями ИОПК-2.1; ИОПК-2.2; ИОПК-2.3; ИОПК-3.3; ИОПК-5.1; ИОПК-5.2; ИОПК-5.3; ИОПК-6.1; ИОПК-6.2; ИОПК-6.3 и результатами обучения:

№	Этапы формирования компетенций (разделы дисциплины)	Код и наименование результатов обучения
1.	Основы программирования на языке Паскаль	ОР-2.1.3. Знает язык программирования Паскаль и основы языка C++. ОР-2.2.1. Знает приемы работы в средах разработки программ Lazarus (свободное ПО) и MS Visual Studio. ОР-2.3.1. Умеет использовать современные инструментальные средства для разработки программных решений. ОР-5.1.1. Знает порядок инсталляции инструментальных систем для разработки приложений. ОР-5.2.1. Умеет инсталлировать программное обеспечение для выполнения разработки программ.
2.	Тестирование и отладка программ	ОР-2.2.1. Знает приемы работы в средах разработки программ Lazarus (свободное ПО) и MS Visual Studio. ОР-2.3.1. Умеет использовать современные инструментальные средства для разработки программных решений. ОР-6.1.1. Знает приемы настройки программного обеспечения. ОР-6.2.2. Умеет разрабатывать, тестировать и отлаживать программы умеренной сложности.
3.	Доказательство свойств программ	ОР-2.1.1. Знает базовые принципы теорий, связанных с

		фундаментальной информатикой и информационными технологиями. ОР-6.2.1. Умеет оценивать эффективность разрабатываемых алгоритмов.
4.	Основные алгоритмы и их трудоемкость	ОР-6.2.1. Умеет оценивать эффективность разрабатываемых алгоритмов. ОР-6.2.2. Умеет разрабатывать, тестировать и отлаживать программы умеренной сложности.
5.	Простые алгоритмы сортировки и поиска и их трудоемкость	ОР-6.2.1. Умеет оценивать эффективность разрабатываемых алгоритмов. ОР-6.2.2. Умеет разрабатывать, тестировать и отлаживать программы умеренной сложности.
6.	Простые рекурсивные алгоритмы	ОР-6.2.1. Умеет оценивать эффективность разрабатываемых алгоритмов. ОР-6.2.2. Умеет разрабатывать, тестировать и отлаживать программы умеренной сложности.
7.	Файлы в Паскале. Взаимодействие с операционной системой	ОР-2.1.3. Знает язык программирования Паскаль и основы языка C++. ОР-2.2.1. Знает приемы работы в средах разработки программ Lazarus (свободное ПО) и MS Visual Studio. ОР-5.3.1. Умеет использовать современные программные системы обеспечения компьютерной безопасности. ОР-6.1.1. Знает приемы настройки программного обеспечения.
8.	Списочные структуры	ОР-6.2.2. Умеет разрабатывать, тестировать и отлаживать программы умеренной сложности.
9.	Рекурсивные алгоритмы бэктрекинга	ОР-6.2.1. Умеет оценивать эффективность разрабатываемых алгоритмов. ОР-6.2.2. Умеет разрабатывать, тестировать и отлаживать программы умеренной сложности.
10.	Алгоритмы над множествами	ОР-2.1.2. Знает принципы разработки алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей. ОР-6.2.1. Умеет оценивать эффективность разрабатываемых алгоритмов. ОР-6.2.2. Умеет разрабатывать, тестировать и отлаживать программы умеренной сложности.
11.	Алгоритмы со строками и таблицами	ОР-2.1.2. Знает принципы разработки алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей.

		<p>ОР-6.2.1. Умеет оценивать эффективность разрабатываемых алгоритмов.</p> <p>ОР-6.2.2. Умеет разрабатывать, тестировать и отлаживать программы умеренной сложности.</p>
12.	Основы программирования на языке Си	<p>ОР-2.1.3. Знает язык программирования Паскаль и основы языка C++.</p> <p>ОР-2.2.1. Знает приемы работы в средах разработки программ Lazarus (свободное ПО) и MS Visual Studio.</p> <p>ОР-2.3.1. Умеет использовать современные инструментальные средства для разработки программных решений.</p> <p>ОР-5.2.1. Умеет устанавливать программное обеспечение для выполнения разработки программ.</p> <p>ОР-6.1.1. Знает приемы настройки программного обеспечения.</p> <p>ОР-3.3.1. Умеет использовать современные информационные технологии, в том числе отечественного производства на всех этапах разработки программных систем</p>
13.	Простые программы на Си	<p>ОР-2.1.3. Знает язык программирования Паскаль и основы языка C++.</p> <p>ОР-2.3.1. Умеет использовать современные инструментальные средства для разработки программных решений.</p> <p>ОР-5.1.1. Знает порядок инсталляции инструментальных систем для разработки приложений.</p>
14.	Синтаксис и семантика языка программирования	<p>ОР-2.1.1. Знает базовые принципы теорий, связанных с фундаментальной информатикой и информационными технологиями.</p>
15.	Алгоритмы с векторами и матрицами	<p>ОР-2.1.2. Знает принципы разработки алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей.</p> <p>ОР-6.2.1. Умеет оценивать эффективность разрабатываемых алгоритмов.</p> <p>ОР-6.2.2. Умеет разрабатывать, тестировать и отлаживать программы умеренной сложности.</p>
16.	Простые алгоритмы над графами	<p>ОР-2.1.2. Знает принципы разработки алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей.</p> <p>ОР-6.2.1. Умеет оценивать эффективность разрабатываемых алгоритмов.</p>

		ОР-6.2.2. Умеет разрабатывать, тестировать и отлаживать программы умеренной сложности.
17.	Циклы и пути в графах	ОР-2.1.2. Знает принципы разработки алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей. ОР-6.2.1. Умеет оценивать эффективность разрабатываемых алгоритмов. ОР-6.2.2. Умеет разрабатывать, тестировать и отлаживать программы умеренной сложности.
18.	Разработка больших программ	ОР-2.1.1. Знает базовые принципы теорий, связанных с фундаментальной информатикой и информационными технологиями. ОР-5.3.1. Умеет использовать современные программные системы обеспечения компьютерной безопасности.

Оценка за промежуточную аттестацию в каждом семестре вычисляется на основе суммы баллов по четырем письменным контрольным работам и сданным лабораторным работам. Таблица перевода оценок из 200-балльной шкалы в 5-балльную:

Баллы -> оценки (итог)		
От	До	
173		отлично
112	172	хорошо
67	111	удовлетворительно
0	66	неудовлетворительн о

Условие получения удовлетворительной оценки – выполнение всех контрольных работ и обязательное выполнение 4-5 определенных заданий в семестре.

### 11. Учебно-методическое обеспечение

а) Электронные учебные курсы по дисциплине в электронном университете «Moodle» <https://moodle.tsu.ru/course/view.php?id=11197>  
<https://moodle.tsu.ru/course/view.php?id=13334>

б) Оценочные материалы текущего контроля и промежуточной аттестации по дисциплине приведены в электронных курсах дисциплины в системе Moodle.

в) План лабораторных занятий по дисциплине приведен в электронных курсах дисциплины в системе Moodle.

г) Методические указания по проведению лабораторных работ опубликованы в электронных курсах дисциплины в системе Moodle.

д) Методические указания по организации самостоятельной работы студентов опубликованы в электронных курсах дисциплины в системе Moodle.



## 12. Перечень учебной литературы и ресурсов сети Интернет

а) основная литература:

- Вирт Н. Алгоритмы + структуры данных = программы. – М.: Мир, 1985. – 406 с.
- Захаров Д.В. Системное программирование. Учебное пособие. – Томск: Изд-во НТЛ, 2007. – 261 с.
- Костюк Ю.Л. Лекции по основам программирования. Учебное пособие. – Томск: Издательский дом ТГУ, 2019. – 259 с.

б) дополнительная литература:

- Страуструп Б. Язык программирования C++. Специальное издание. – М.: Бином, 2017. - 1136 с.

в) ресурсы сети Интернет:

- Информационный портал для разработчиков на Free Pascal & Lazarus. – <http://www.freepascal.ru/>

## 13. Перечень информационных технологий

а) лицензионное и свободно распространяемое программное обеспечение:

- Microsoft Office Standart 2013 Russian: пакет программ. Включает приложения: MS Office Word, MS Office Excel, MS Office PowerPoint, MS Office OneNote, MS Office Publisher, MS Outlook, MS Office Web Apps (Word Excel MS PowerPoint Outlook);
- MS Visual Studio 2019 и выше;
- Lazarus v 1.8.2 и выше.

б) информационные справочные системы:

- Электронный каталог Научной библиотеки ТГУ – <http://chamo.lib.tsu.ru/search/query?locale=ru&theme=system>
- Электронная библиотека (репозиторий) ТГУ – <http://vital.lib.tsu.ru/vital/access/manager/Index>
- ЭБС Лань – <http://e.lanbook.com/>
- Учебники по Lazarus и Pascal – [http://wiki.freepascal.org/Lazarus\\_Documentation/ru](http://wiki.freepascal.org/Lazarus_Documentation/ru)
- Справочник по языку C++. – <https://msdn.microsoft.com/ru-ru/library/3bstk3k5.aspx>
- Справочник по C/C++. – <http://mycpp.ru/cpp/scpp/>

## 14. Материально-техническое обеспечение

Для материально-технического обеспечения дисциплины требуется наличие компьютерной техники с установленным соответствующим программным обеспечением и оборудования, поддерживающего проведение презентаций, выход в сеть Интернет.

## 15. Информация о разработчиках

Ю.Л. Костюк, д-р техн. наук, профессор, кафедра теоретических основ информатики ТГУ, профессор.

И.Л. Фукс, кафедра теоретических основ информатики ТГУ, старший преподаватель.