

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

Механико-математический факультет

УТВЕРЖДАЮ:
Декан ММФ ТГУ
Л. В. Гензе

Рабочая программа дисциплины

Программирование

по направлению подготовки

**01.03.01 Математика, 01.03.03 Механика и математическое моделирование,
02.03.01 Математика и компьютерные науки**

Направленность (профиль) подготовки:

**Современная математика и математическое моделирование
Теоретическая, вычислительная и экспериментальная механика
Вычислительная математика и компьютерное моделирование**

Форма обучения

Очная

Квалификация

Бакалавр

Год приема

2024

СОГЛАСОВАНО:
Руководитель ОП
Л.В. Гензе

Председатель УМК
Е.А. Тарасов

Томск – 2024

1. Цель и планируемые результаты освоения дисциплины

Целью освоения дисциплины является формирование следующих компетенций:

ОПК-2. Способен разрабатывать, анализировать и внедрять математические модели в современной науке и технике, экономике и управлении

ОПК-3. Способен разрабатывать и реализовывать алгоритмы и компьютерные программы, пригодные для практического применения

Результатами освоения дисциплины являются следующие:

РООПК 2.1 – Проводит поиск, сбор и структурирование/систематизацию информации по заданным критериям для построения математической модели, используемой для решения задачи.

РООПК 2.2 – Применяет стандартные и типовые действия при проведении всех этапов математического моделирования. построении математической модели определенного типа

РООПК 2.3 – Применяет подходы визуализации и представления результатов математического моделирования для апробации и демонстрации в виде отчетов, презентаций и научных текстов

РООПК 3.1. Владеет теоретическими основами программирования и алгоритмизации

РООПК 3.2. Способен реализовывать алгоритмы на языках программирования высокого уровня

РООПК 3.3 Способен разрабатывать программные алгоритмы при решении задач методами математики и механики

2. Задачи освоения дисциплины

- Сформировать у обучающихся систему понятий и представлений об основах программирования;
- Изучить основы программирования, освоить методы и процессы решения задач на современных языках программирования, этапы построения программ;
- Выработать навыки составления программ на современных языках программирования для решения различных задач.

3. Место дисциплины в структуре образовательной программы

Дисциплина относится к Блоку 1 «Дисциплина (модули)».

Дисциплина относится к обязательной части образовательной программы.

Дисциплина входит в модуль Модуль: Компьютерные науки.

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

Первый семестр, экзамен

Второй семестр, зачет

Третий 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 12 з.е., 432 часов, из которых:

-лекции: 64 ч.

-практические занятия: 128 ч.

Объем самостоятельной работы студента определен учебным планом.

8. Содержание дисциплины, структурированное по темам

Тема 1. Теоретические основы программирования и алгоритмизации. Язык программирования C++.

Краткое содержание темы: История развития вычислительной техники. Классы современных компьютеров. Архитектура фон Неймана. Принципиальное устройство компьютеров. Оперативная память. Процессор. Система элементарных команд процессора. Понятие алгоритма, его свойства. Способы представления алгоритмов. Представление алгоритмов с помощью блок-схем. Позиционные системы счисления. Структура программы на языке C (C++). Понятие типа данных. Стандартные скалярные типы данных. Переменные в языке C (C++), их описание и использование; область действия переменных. Выражения. Оператор присваивания. Операторы ввода-вывода с использованием стандартных устройств в языках C, C++. Форматированный ввод-вывод в языке C. Условный оператор. Оператор множественного выбора **switch**. Операторы цикла с пред-, и постусловием. Тестирование и отладка программ. Приемы, часто применяемые при отладке. Основные принципы структурного программирования. Теорема БомеДжакопини. Метод проектирования программ «сверху вниз» – пошаговая детализация программ. Логические утверждения как характеристика множества допустимых значений переменных. Элементы теории Т. Хоара: определение частичной правильности последовательности операторов, аксиома правильности для оператора присваивания, теорема о достаточном признаке частичной правильности цикла с предусловием. понятия инварианта цикла и ограничивающей функции. Массивы как тип данных. Одномерные, двумерные массивы. Оператор цикла **for**. Указатели и их использование. Динамические переменные. Эквивалентность массивов и указателей. Динамические массивы. Понятие подпрограммы. Описание, вызов подпрограмм. Виды формальных параметров: входные, результаты, модифицируемые. Описание функции в языке C (C++). Формальные параметры функций. Вызов функции в языке C (C++). Действия компьютера при вызове функции. Соответствие формальных и фактических параметров при вызове функций. Глобальные параметры. Использование указателей и ссылок в качестве параметров функций. Использование массивов в качестве параметров функций. Указатели на функции. Прототипы функций. Заголовочные файлы и их подключение к проекту. Перегрузка функций. Формальные параметры функций по умолчанию.

Тема 2. Структуры данных, строки, функции, файлы на языке C++. Алгоритмы поиска, сортировки, хеширования, оценка их эффективности и сложности

Краткое содержание темы: Структуры данных, строки, функции, файлы на языке C++. Рекурсивные функции. Алгоритм поиска с возвратом на примере поиска пути в дорожной сети. Эффективность и сложность алгоритмов. Классификация алгоритмов по их сложности. Работа с информацией, расположенной в таблице (одномерном массиве). Постановка задач поиска, вставки и удаления элементов таблицы. Алгоритмы поиска, вставки и удаления в информации в неупорядоченном массиве. Оценка их эффективности. Алгоритм бинарного поиска в упорядоченном массиве; оценка его эффективности. Алгоритмы вставки и удаления информации в упорядоченном массиве, оценка их эффективности. Постановка задачи сортировки одномерного массива. Алгоритмы

сортировки: метод «пузырька», метод простого включения, метод прямого выбора, метод быстрой сортировки. Оценка их эффективности. Повышение эффективности стандартных методов сортировки. Хэширование как метод размещения данных в массиве. Коллизии и способы их разрешения. Алгоритмы поиска и вставки информации в массив при использовании хэширования. Преимущества и недостатки технологии хэширования.

Тема 3. Динамические структуры данных: стек, очередь, список, деревья. Объектно-ориентированное программирование

Краткое содержание темы: Алгоритм поиска с возвратом на примере поиска пути в дорожной сети. Стек как структура данных. Логическое описание. Варианты физической реализации стека: на базе массива и с помощью динамических переменных. Функции, реализующие операции со стеком. Очередь как структура данных. Логическое описание. Варианты реализации очереди с помощью массива и динамических переменных. Функции, реализующие операции с очередью. Линейный однонаправленный список (ЛОС), логическое описание. Варианты реализации списка с помощью массива и динамических переменных. Функции, реализующие операции с ЛОС. Деревья как структуры данных. Основные определения и способы реализации. Обходы деревьев. Деревья бинарного поиска. Алгоритмы поиска, вставки и удаления элементов в бинарном дереве поиска; оценка их эффективности. Начала объектно-ориентированного программирования (ООП). Классы как обобщение понятия структуры. Инкапсуляция – соединение переменных и методов работы с ними в одной структуре. Описание класса в языке C++. Конструкторы классов; конструкторы по умолчанию, конструкторы копирования. Шаблоны классов. Примеры создания классов для реализации стека, очереди и ЛОС. Использование принципа наследования при создании иерархии классов. Стандартные классы из библиотеки C++: строки, потоки, контейнеры. Понятие операционной системы, основные элементы операционной системы, принципы взаимодействия прикладных программ с операционными системами. Исключения как классы, обеспечивающие взаимодействие программы с операционной системой.

Тема 4. Язык программирования Python. Внешние модули

Введение в среду программирования Python. Основы синтаксиса, типы данных, операторы, условные операторы, циклы, встроенные функции. Пользовательские функции, стандартная и нестандартная библиотека Python, обработка исключений, работа с файлами, типы данных (продолжение). Решение классических задач программирования для освоения основ синтаксиса языка Python. Знакомство с IDE Jupyter Notebook, Google Colab. Первый запуск, основные элементы интерфейса, работа с ячейками, клавиши быстрого доступа. Ноутбук: работа как и с ячейками кода, так и с текстовыми ячейками, основы языка Markdown.

Модуль numpy. Работа с массивами, линейная алгебра, работа с полиномами, генерация случайных чисел. Модуль matplotlib. Основные элементы графика, компоновка нескольких графиков, типы графиков, задание цвета и цветовой схемы, цветовая полоса, анимация (по желанию). Модуль pandas. Структура данных pandas, чтение и запись, работа со структурами pandas, работа с датами, вычисление статистик, визуализация – начало анализа данных. Модуль scipy. Статистическая кластеризация, численное интегрирование, решение краевой задачи и задачи Коши для ОДУ и систем, интерполяция, оптимизация, работа с пространственными структурами данных, статистика. Параллельные вычисления с dask, символьные вычисления с sympy.

9. Текущий контроль по дисциплине

Текущий контроль по дисциплине проводится путем контроля посещаемости, проведения контрольных работ, тестов по лекционному материалу, выполнения индивидуальных заданий и фиксируется в форме контрольной точки не менее одного раза в семестр.

Текущий контроль качества, объема и получения компетенций проводится во время консультации с преподавателем, а также на практических занятиях при выполнении индивидуальных практических занятий. Практические занятия по дисциплине «Программирование» призваны закрепить знания бакалавра по отдельным разделам курса, привить им навыки свободного владения технологиями программирования и развить алгоритмическое мышление. На практических занятиях обучающийся разрабатывает алгоритмы решения задач, составляет программы с использованием языка программирования С++ и тестирует программы выполненных заданий. По результатам выполнения индивидуального задания пишется отчет. В таблице 1 приведена система оценивания индивидуального задания.

Таблица 1. Система критериев при оценивании индивидуального задания

Критерии соответствия	Оценка
Содержание отчета и ответы по индивидуальному заданию являются полными. Студент правильно понимает терминологию. Демонстрирует умение понимать, доказательно и логически связно отвечать на вопросы.	зачтено
Неполное, логически противоречивое изложение отчета. Студент не понимает и неправильно использует терминологию. Не может доказательно и логически связно отвечать на вопросы. Отчет вообще не подготовлен к защите.	не зачтено

При ответе на вопросы при защите отчета по индивидуальному заданию оценивается полнота и точность ответа, логичность и аргументированность изложения материала, умения использовать в ответе фактический материал. Для выставления текущей успеваемости при контроле СРС рекомендуется использовать следующую таблицу (Таблица 1).

10. Учебно-методическое обеспечение

а) Электронный учебный курс по дисциплине в электронном университете «Lms» - <https://lms.tsu.ru/course/view.php?id=8367> (1 курс 1 семестр)
<https://lms.tsu.ru/course/view.php?id=13057> (1 курс 2 семестр)
<https://lms.tsu.ru/course/view.php?id=8378> (2 курс 1 семестр)

б) Оценочные материалы текущего контроля и промежуточной аттестации по дисциплине. Находятся на сайте <https://tsu.ru/sveden/education/eduop/>

- в) План семинарских / практических занятий по дисциплине.
- г) Методические указания по проведению лабораторных работ.
- д) Методические указания по организации самостоятельной работы студентов.

В ходе реализации дисциплины используются классические образовательные технологии – лекции, практические занятия; самостоятельное изучение рекомендованной литературы и постепенное выполнение индивидуального задания; промежуточная аттестация в приема зачетов (2, 3 семестры) и экзаменов (1,4 семестры).

Самостоятельная работа включает в себя: теоретическое освоение лекционного курса, практическое выполнение контрольных заданий и индивидуальных заданий, подготовку к зачету и экзамену. Для выполнения самостоятельной работы обеспечивается доступ к информационным ресурсам курса:

- материалы лекций;
- список вопросов для самостоятельной проверки знаний и подготовки к экзамену.

- список литературы, включающий учебники и книги по изучаемым в курсовопросам.

Все контрольные работы и индивидуальные задания подобраны так, чтобы максимально стимулировать психологическую установку студентов-математиков на формирование связи между математической теорией и ее практическим применением. Отчет по каждой лабораторной работе включает теоретическую часть, выполненное практическое задание и анализ полученных результатов.

Для успешного освоения материала студентам необходимо посещать занятия, а во время самостоятельной работы пользоваться основной и дополнительной литературой, базами данных и информационно-справочными системами, которые представлены в списке литературы. Самостоятельная работа студентов состоит в повторении материала с практических занятий и самостоятельного изучения дополнительных вопросов, более глубокого анализа темы с помощью литературы.

11. Перечень учебной литературы и ресурсов сети Интернет

а) основная литература:

1. Керниган Б., Ритчи Д. Язык программирования Си. СПб.: «Невский диалект», 2001.
2. Т. А. Павловская. С, С++. Программирование на языке высокого уровня. СПб.: ПИТЕР, 2002.
3. В. В. Подбельский Язык С++. М.: Финансы и статистика, 1999.
4. Т. А. Павловская, Ю. А. Щупак. С, С++. Структурное программирование: Практикум. СПб.:ПИТЕР, 2002.
5. Васильев А.Н. Программирование на С++ в примерах и задачах.
6. Конова Е., Поллак Г. Алгоритмы и программы. С++. СПб.: Издательство «Лань», 2017.384 с.
7. Ульман Дж. Д., Уидом Дж. Введение с системы баз данных. - М.: Издательство «Лори», 2000.
8. Ю.Б. Бакаревич, Н.В.Пушкина. Самоучитель MS Office 2016. – СПб., БХВПетербург, 2017, 418 с
9. Любанович Б. Простой Python. Современный стиль программирования. 2-е. изд. – СПб.: Питер, 2021. – 592 с.
10. Чан. Дж. Python: быстрый старт. — СПб.: Питер, 2021. — 224 с.
11. Стивенсон Б. Python. Сборник упражнений / пер. с англ. А. Ю. Гинько. – М.: ДМК Пресс, 2021. – 238 с.
12. Лутц М.. Python. Карманный справочник. – М.: Вильямс, 2019. – 320 с.
13. Вабищевич Н. П. Численные методы. Вычислительный практикум. Практическое применение численных методов при использовании алгоритмического языка Python. – Москва: USSR, 2020. – 320 с.
14. Хайбрахманов С. А. Основы научных расчётов на языке программирования Python : УП. — Челябинск : Изд-во Челяб. гос. ун-та, 2019. — 96 с.
15. Криволапов С. Я., Хрипунова М. Б. Математика на Python: учебник. – Москва: КНОРУС, 2022. – 456 с.

б) дополнительная литература:

1. Вопросы и ответы по С и С++. Москва: «МикроАрт», 1997.
2. Брайан Оверленд. С++ без страха. М.; изд-во «Триумф», 2005.
3. Йодан Э. Структурное программирование и конструирование программ. М.:Мир, 1979.
4. Дейкстра. Дисциплина программирования. М.: Мир, 1978.

5. Вирт Н. Алгоритмы и структуры данных. М: Мир, 1985.
6. Мейер В., Бодуэн К. Методы программирования. М.:Мир,1982, Т.1,2.
7. Могилев А.В., Пак Н.И., Хеннер Е.К. Информатика. М.- Академия, 1999.
8. Дейт К. Дж. Введение в системы баз данных. – К.; М.; СПб.: Издательский дом «Вильямс», 1999
9. Задачник по программированию для математиков. Томск: Изд-во Том. ун-та, 2001.
10. Страуструп Б. Программирование. Принципы и практика использования С++. Москва:Вильямс , 2011 – 1238 стр.
11. Дейтел П., Дейтел Х. Python: Искусственный интеллект, большие данные и облачные вычисления. — СПб.: Питер, 2020. — 864 с.
12. Нуньес-Иглесиас Х., Уолт ван дер Ш., Дэшноу Х. Элегантный SciPy. Научное программирование на Python. – М. ДМК Пресс, 2018. – 266 с.
13. Брюс, П. Практическая статистика для специалистов Data Science. 2-е. изд., перераб. и доп. – СПб.: БХВ-Петербург, 2021. – 352 с.
14. Кристиан Хилл. Научное программирование на Python / пер. с англ. А. В. Снастина. – М.: ДМК Пресс, 2021. – 646 с.

в) ресурсы сети Интернет:

1. ЭБС Лань <http://e.lanbook.com/books>
2. Открытый университет Интуит.ру <http://intuit.ru>;
3. <http://www.intuit.ru/studies/courses/2260/156/info>
4. <http://www.intuit.ru/studies/courses/66/66/info>
5. Богословский Н.Н. Программирование на С++ Электронный ресурс: учебнометодический комплекс. Томск : [ИДО ТГУ] , 2015 URL: <http://lms.tsu.ru/course/view.php?id=431>
6. Курс на coorsera <https://www.coursera.org/learn/python-osnovy-programmirovaniya>
7. Курс на sololearn <https://www.sololearn.com/learning/1073>

12. 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 On-eNote, MS Office Publisher, MS Outlook, MS Office Web Apps (Word Excel MS PowerPoint Outlook); - интегрированная среда для работы на С/С++.
– публично доступные облачные технологии (Google Docs, Яндекс диск и т.п.).

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/>
- ЭБС Консультант студента – <http://www.studentlibrary.ru/>
- Образовательная платформа Юрайт – <https://urait.ru/>
- ЭБС ZNANIUM.com – <https://znanium.com/>
- ЭБС IPRbooks – <http://www.iprbookshop.ru/>

13. Материально-техническое обеспечение

Аудитории для проведения занятий лекционного типа.

Аудитории для проведения занятий семинарского типа, индивидуальных и групповых консультаций, текущего контроля и промежуточной аттестации.

Помещения для самостоятельной работы, оснащенные компьютерной техникой и доступом к сети Интернет, в электронную информационно-образовательную среду и к информационным справочным системам.

14. Информация о разработчиках

Старченко Александр Васильевич, профессор, доктор физико-математических наук.

Гольдин Виктор Данилович, доцент, кандидат физико-математических наук

Зюзьков Валентин Михайлович, доцент, кандидат физико-математических наук

Лаева Валентина Ивановна, старший преподаватель

Данилкин Евгений Александрович, доцент, кандидат физико-математических наук

Шельмина Елена Александровна, доцент, кандидат физико-математических наук

Стребкова Екатерина Александровна, старший преподаватель

Михайлов Михаил Дмитриевич, старший преподаватель