

**МИНИСТЕРСТВО ТРАНСПОРТА РОССИЙСКОЙ ФЕДЕРАЦИИ**  
**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ**  
**УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**  
**«РОССИЙСКИЙ УНИВЕРСИТЕТ ТРАНСПОРТА»**  
**(РУТ (МИИТ))**



Рабочая программа дисциплины (модуля),  
как компонент образовательной программы  
высшего образования - программы бакалавриата  
по направлению подготовки  
01.03.02 Прикладная математика и информатика,  
утвержденной первым проректором РУТ (МИИТ)  
Тимониным В.С.

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)**

**Программирование**

Направление подготовки: 01.03.02 Прикладная математика и информатика

Направленность (профиль): Математическое моделирование и системный анализ

Форма обучения: Очная

Рабочая программа дисциплины (модуля) в виде электронного документа выгружена из единой корпоративной информационной системы управления университетом и соответствует оригиналу

Простая электронная подпись, выданная РУТ (МИИТ)  
ID подписи: 5665  
Подписал: заведующий кафедрой Нутович Вероника Евгеньевна  
Дата: 02.09.2024

## 1. Общие сведения о дисциплине (модуле).

Целью освоения дисциплины (модуля) является:

- изучение основ прикладного и системного программирования, средств создания программного обеспечения для решения научных, прикладных, деловых, математических и др. задач. В качестве языка программирования приняты язык C++, реализованный в интегрированной среде Microsoft Visual Studio 2022.

Задачами освоения дисциплины (модуля) являются:

- обучение студента основным понятиям в области программирования на языке C++;
- приобретение навыков разработки алгоритмов и написания программ на языке C++ для практического использования.

## 2. Планируемые результаты обучения по дисциплине (модулю).

Перечень формируемых результатов освоения образовательной программы (компетенций) в результате обучения по дисциплине (модулю):

**ПК-3** - Уметь разрабатывать методики выполнения аналитических работ; планировать, организовывать и контролировать аналитические работы в информационно-технологическом проекте.

Обучение по дисциплине (модулю) предполагает, что по его результатам обучающийся будет:

### **Знать:**

- типовые решения, библиотеки программных модулей, шаблоны, классы объектов, используемые при разработке программного обеспечения;
- методы и средства проектирования программного обеспечения.

### **Уметь:**

- создавать блок-схемы алгоритмов функционирования разрабатываемых программных продуктов;
- применять язык программирования для написания программного кода;
- использовать стандартные библиотеки для написания программного обеспечения;
- осуществлять отладку программных продуктов.

### **Владеть:**

- навыками изучения технической документации по языку программирования;
- методами создания программных продуктов в соответствии с

техническим заданием;

- навыками разработки эксплуатационной документации на разработанный программный продукт.

### 3. Объем дисциплины (модуля).

#### 3.1. Общая трудоемкость дисциплины (модуля).

Общая трудоемкость дисциплины (модуля) составляет 7 з.е. (252 академических часа(ов)).

3.2. Объем дисциплины (модуля) в форме контактной работы обучающихся с педагогическими работниками и (или) лицами, привлекаемыми к реализации образовательной программы на иных условиях, при проведении учебных занятий:

Тип учебных занятий	Количество часов		
	Всего	Семестр	
		№2	№3
Контактная работа при проведении учебных занятий (всего):	176	80	96
В том числе:			
Занятия лекционного типа	80	32	48
Занятия семинарского типа	96	48	48

3.3. Объем дисциплины (модуля) в форме самостоятельной работы обучающихся, а также в форме контактной работы обучающихся с педагогическими работниками и (или) лицами, привлекаемыми к реализации образовательной программы на иных условиях, при проведении промежуточной аттестации составляет 76 академических часа (ов).

3.4. При обучении по индивидуальному учебному плану, в том числе при ускоренном обучении, объем дисциплины (модуля) может быть реализован полностью в форме самостоятельной работы обучающихся, а также в форме контактной работы обучающихся с педагогическими работниками и (или) лицами, привлекаемыми к реализации образовательной программы на иных условиях, при проведении промежуточной аттестации.

### 4. Содержание дисциплины (модуля).

#### 4.1. Занятия лекционного типа.

№ п/п	Тематика лекционных занятий / краткое содержание
1	<b>Алгоритмизация</b> Рассматриваемые вопросы: - понятие алгоритма; - основные понятия алгоритмизации; - формы записи алгоритмов; - основные алгоритмические конструкции; - блок-схемы.
2	<b>Основы программирования</b> Рассматриваемые вопросы: - лексика языка; - переменные и константы; - типы данных; - выражения и операции.
3	<b>Основы программирования</b> Рассматриваемые вопросы: - логические конструкции и ветвления; - конструкция if/else; - вложенные условия; - конструкция switch/case; - циклы.
4	<b>Составные типы данных</b> Рассматриваемые вопросы: - работа с массивами; - указатели; - массивы и указатели; - массивы символов; - структуры.
5	<b>Процедурное программирование</b> Рассматриваемые вопросы: - функции; - перегрузка функций; - способы передачи параметров функциям.
6	<b>Основные особенности ООП</b> Рассматриваемые вопросы: - абстрагирование, существенные свойства объектов; - свойства класса объектов; - отношения между классами объектов: ассоциация, наследование, агрегация, зависимость; - описание состояния и поведение в классе объектов; - поля и методы класса объектов; - состояние и индивидуальность объекта.
7	<b>Инкапсуляция</b> Рассматриваемые вопросы: - определение инкапсуляции классов объектов; - время жизни и область видимости; - локальность и глобальность; - приватная область; - защищенная область; - общедоступная область.
8	<b>Объекты классов</b> Рассматриваемые вопросы:

№ п/п	Тематика лекционных занятий / краткое содержание
	<ul style="list-style-type: none"> <li>- объекты и классы;</li> <li>- доступ к членам класса;</li> <li>- возвращение объектов функциями;</li> <li>- работа с указателем this;</li> <li>- массивы объектов;</li> <li>- указатели на объекты.</li> </ul>
9	<p><b>Специальные методы в классе объектов</b></p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> <li>- конструкторы и деструкторы;</li> <li>- создание конструкторов по умолчанию;</li> <li>- виртуальные деструкторы;</li> <li>- последовательность вызовов конструкторов и деструкторов;</li> <li>- геттеры и сеттеры;</li> <li>- дружественные функции.</li> </ul>
10	<p><b>Перегрузка операторов</b></p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> <li>- перегрузка операторов с использованием функций-членов;</li> <li>- перегрузка унарных операторов;</li> <li>- перегрузка операторов с использованием функций-не членов класса;</li> <li>- перегрузка унарных операторов с использованием дружественных функций;</li> <li>- перегрузка оператора индексации массивов ([]);</li> <li>- перегрузка оператора "()".</li> </ul>
11	<p><b>Наследование</b></p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> <li>- понятие о наследовании;</li> <li>- управление доступом к членам базового класса;</li> <li>- использование защищенных членов класса;</li> <li>- спецификатор protected для наследования базового класса;</li> <li>- наследование нескольких базовых классов;</li> <li>- конструкторы, деструкторы и наследование;</li> <li>- передача параметров конструкторам базового класса;</li> <li>- предоставление доступа.</li> </ul>
12	<p><b>Виртуальные функции и полиморфизм</b></p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> <li>- определение полиморфизма классов объектов;</li> <li>- указатели на производные типы;</li> <li>- виртуальные функции;</li> <li>- наследование виртуальных функций;</li> <li>- чисто виртуальные функции и абстрактные классы;</li> <li>- раннее и позднее связывание.</li> </ul>
13	<p><b>Контейнер вектор</b></p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> <li>- вектор как динамический массив;</li> <li>- функции-члены класса vector;</li> <li>- использование итератора для доступа к вектору;</li> <li>- вставка и удаление элементов из вектора.</li> </ul>
14	<p><b>Контейнер списки</b></p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> <li>- функции-члены класса list;</li> <li>- сортировка списка;</li> <li>- объединение списков.</li> </ul>

№ п/п	Тематика лекционных занятий / краткое содержание
15	Алгоритмы/ Рассматриваемые вопросы - алгоритмы ST; - подсчет элементов; - удаление и замена элементов; - преобразование последовательности.
16	Процессор C++ Рассматриваемые вопросы: - директива; - макроопределения, действующие как функции; - директива #include; - директивы условной компиляции; - Операторы препроцессора "#" и "##".
17	Система ввода-вывода в C++ Рассматриваемые вопросы: - потоки и буферы; - встроенные потоки; - форматирование вывода; - установка ширины поля, точности и символов заполнения; - манипуляторы ввода-вывода.
18	Система ввода-вывода в C++ Рассматриваемые вопросы: - файловый ввод-вывод; - чтение и запись текстовых файлов; - функции get() и put(); - работа в режиме командной строки.
19	Обработка исключений Рассматриваемые вопросы: - основы обработки исключительных ситуаций; - функции exit() и abort(); - перехват исключений классового типа; - перехват исключений базового класса; - исключение bad_alloc и оператор new.
20	Обработка исключений Рассматриваемые вопросы: - перехват всех исключений; - повторное генерирование исключений; - обработка исключений, сгенерированных оператором new; - перегрузка операторов new и delete; - перегрузка nothrow.

#### 4.2. Занятия семинарского типа.

##### Лабораторные работы

№ п/п	Наименование лабораторных работ / краткое содержание
1	Алгоритмизация В результате выполнения лабораторной работы студент получает навык составления блок-схем.
2	Основы программирования

№ п/п	Наименование лабораторных работ / краткое содержание
	В результате выполнения лабораторной работы студент получает навык работы с базовыми конструкциями, типами данных и синтаксическими особенностями языка C++.
3	<b>Реализация ветвящихся процессов</b> В результате выполнения лабораторной работы студент получает навык работы с оператором if/else на языке C++.
4	<b>Реализация ветвящихся процессов</b> В результате выполнения лабораторной работы студент получает навык работы с оператором switch/case на языке C++.
5	<b>Реализация циклических процессов</b> В результате выполнения лабораторной работы студент получает навык работы с операторами циклов (while, do while, for) на языке C++.
6	<b>Реализация циклических процессов</b> В результате выполнения лабораторной работы студент получает навык работы с вложенными операторами циклов на языке C++.
7	<b>Составные типы данных</b> В результате выполнения лабораторной работы студент получает навык работы с массивами на языке C++.
8	<b>Строковые типы данных</b> В результате выполнения лабораторной работы студент получает навык со строковыми переменными и основными библиотечными функциями обработки строк. на языке C++.
9	<b>Строковые типы данных</b> В результате выполнения лабораторной работы студент получает навык работы с массивами строк на языке C++.
10	<b>файловый ввод/вывод</b> В результате выполнения лабораторной работы студент получает навык работы с файловой системой и операциями файлового ввода/вывода. на языке C++.
11	<b>Процедурное программирование</b> В результате выполнения лабораторной работы студент получает навык работы с функциями, передачей и возвращением параметров.
12	<b>Составные типы данных</b> В результате выполнения лабораторной работы студент получает навык работы со структурами на языке C++
13	<b>Сортировка данных</b> В результате выполнения лабораторной работы студент получает навык работы с сортировкой одномерных массивов на языке C++
14	<b>Сортировка данных</b> В результате выполнения лабораторной работы студент получает навык работы с сортировкой двумерных массивов на языке C++
15	<b>Сортировка данных</b> В результате выполнения лабораторной работы студент получает навык работы с сортировкой массивов строк на языке C++
16	<b>Объектно-ориентированное программирование.</b> В результате выполнения лабораторной работы студент получает навык навыки разработки алгоритмов программ с применением объектно-ориентированной подхода на языке C++.
17	<b>C++ Конструкторы и деструкторы.</b> В результате выполнения лабораторной работы студент получает навыки работы с разными типами конструкторов и деструкторами
18	<b>C++ Конструкторы и деструкторы.</b>

№ п/п	Наименование лабораторных работ / краткое содержание
	В результате выполнения лабораторной работы студент получает навыки работы с перегруженными конструкторами и виртуализацией деструкторов.
19	С++ Наследование. В результате выполнения лабораторной работы студент получает навыки наследование одного класса от другого.
20	С++ Полиморфизм. В результате выполнения лабораторной работы студент получает навыки переопределения методов в наследуемых классах.
21	С++ Обработка исключений. В результате выполнения лабораторной работы студент получает навык разработки алгоритмов программ с использованием технологий обработки исключений на языке С++.
22	С++ Стандартная библиотека шаблонов. В результате выполнения лабораторной работы студент получает навык работы со стандартной библиотекой шаблонов.
23	Строковые типы данных В результате выполнения лабораторной работы студент получает навык работы со строковыми переменными с использованием класса string на языке С++.
24	Динамические массивы В результате выполнения лабораторной работы студент получает навык работы со стандартным шаблоном vector, реализующим динамический массив на языке С++.
25	Списки В результате выполнения лабораторной работы студент получает навык работы со стандартным шаблоном vector, реализующим работу со списками языке С++.

#### 4.3. Самостоятельная работа обучающихся.

№ п/п	Вид самостоятельной работы
1	Работа с литературой.
2	Работа с лекционным материалом.
3	Текущая подготовка к занятиям.
4	Подготовка к промежуточной аттестации.
5	Подготовка к текущему контролю.

#### 5. Перечень изданий, которые рекомендуется использовать при освоении дисциплины (модуля).

№ п/п	Библиографическое описание	Место доступа
1	Огнева, М.В. Программирование на языке С++: практический курс: учебное пособие для вузов / М.В. Огнева, Е.В. Кудрина, А.А.Казачкова. - Москва : Издательство Юрайт, 2024. — 342 с. — (Высшее образование) - ISBN 978-5-534-	<a href="https://urait.ru/viewer/programmirovani-na-yazyke-s-prakticheskij-kurs-555533">https://urait.ru/viewer/programmirovani-na-yazyke-s-prakticheskij-kurs-555533</a>



	18949-0	
2	Барков, И.А. Объектно-ориентированное программирование: учебник для вузов/ И. А. Барков - Санкт-Петербург: Лань, 2023-700 с.-	<a href="https://e.lanbook.com/book/329549">https://e.lanbook.com/book/329549</a>
3	Зоткин, С.П. Программирование на языке высокого уровня C/C++ : учебное пособие/ С. П. Зоткин - Москва: МИСИ - МГСУ , 2018-140с.- ISBN 978-5-7264-1810-0	<a href="https://e.lanbook.com/book/108512">https://e.lanbook.com/book/108512</a>
4	Быков, А.Ю.Решение задач на языках программирования Си и Си++:методическое указание. А.Ю. Быков А.Ю. - Москва: МГТУ им Н.Э. Баумана , 2017-248 с.-	<a href="https://e.lanbook.com/book/103505">https://e.lanbook.com/book/103505</a>
5	Страуструп, Б. Язык программирования C++ для профессионалов: учебное пособие/ Б. Страуструп. – Москва: ИНТУИТ, 2016. – 670 с.	<a href="https://e.lanbook.com/book/100542">https://e.lanbook.com/book/100542</a>

6. Перечень современных профессиональных баз данных и информационных справочных систем, которые могут использоваться при освоении дисциплины (модуля).

- Научно-техническая библиотека РУТ (МИИТ) (<http://library.miit.ru>);
- Информационный портал Научная электронная библиотека eLIBRARY.RU ([www.elibrary.ru](http://www.elibrary.ru));
- Образовательная платформа «Юрайт» (<https://urait.ru/>);
- Электронно-библиотечная система издательства «Лань» (<http://e.lanbook.com/>);
- Электронно-библиотечная система ibooks.ru (<http://ibooks.ru/>).
- Интернет-университет информационных технологий (<http://www.intuit.ru/>).

7. Перечень лицензионного и свободно распространяемого программного обеспечения, в том числе отечественного производства, необходимого для освоения дисциплины (модуля).

- Операционная система Windows;
- Microsoft Office;
- MS Teams;
- Поисковые системы.

8. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю).

Для проведения занятий лекционного типа требуются аудитории,

оснащенные компьютерной техникой и наборами демонстрационного оборудования.

Для лабораторных занятий – наличие персональных компьютеров.

9. Форма промежуточной аттестации:

Зачет во 2, 3 семестрах.

10. Оценочные материалы.

Оценочные материалы, применяемые при проведении промежуточной аттестации, разрабатываются в соответствии с локальным нормативным актом РУТ (МИИТ).

Авторы:

старший преподаватель кафедры  
«Цифровые технологии управления  
транспортными процессами»

Л.В. Александрова

Согласовано:

Заведующий кафедрой ЦТУТП

В.Е. Нутович

Председатель учебно-методической  
комиссии

Н.А. Андриянова