

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 - программа бакалавриата
по направлению подготовки
01.03.02 Прикладная математика и информатика,
утвержденной первым проректором ФГАОУ ВО
РУТ (МИИТ) Тимониним 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 ФГАОУ ВО РУТ
(МИИТ)

ID подписи: 5665

Подписал: заведующий кафедрой Нутович Вероника
Евгеньевна

Дата: 01.09.2026

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 9 з.е. (324 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):	112	32	80
В том числе:			
Занятия лекционного типа	48	16	32
Занятия семинарского типа	64	16	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 212 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	Алгоритмизация Рассматриваемые вопросы: - понятие алгоритма; - основные понятия алгоритмизации; - формы записи алгоритмов; - основные алгоритмические конструкции; - блок-схемы.
2	Основы программирования Рассматриваемые вопросы: - лексика языка; - переменные и константы; - типы данных; - выражения и операции.
3	Основы программирования Рассматриваемые вопросы: - логические конструкции и ветвления; - конструкция if/else; - вложенные условия; - конструкция switch/case; - циклы.
4	Составные типы данных Рассматриваемые вопросы: - работа с массивами; - указатели; - массивы и указатели; - массивы символов; - структуры.
5	Процедурное программирование Рассматриваемые вопросы: - функции; - перегрузка функций; - способы передачи параметров функциям.
6	Основные особенности ООП 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	Инкапсуляция Рассматриваемые вопросы: - определение инкапсуляции классов объектов; - время жизни и область видимости; - локальность и глобальность; - приватная область; - защищенная область; - общедоступная область.
8	Объекты классов Рассматриваемые вопросы:

№ п/п	Тематика лекционных занятий / краткое содержание
	- объекты и классы; - доступ к членам класса; - возвращение объектов функциями; - работа с указателем <code>this</code>; - массивы объектов; - указатели на объекты.
9	Специальные методы в классе объектов Рассматриваемые вопросы: - конструкторы и деструкторы; - создание конструкторов по умолчанию; - виртуальные деструкторы; - последовательность вызовов конструкторов и деструкторов; - геттеры и сеттеры; - дружественные функции.
10	Перегрузка операторов Рассматриваемые вопросы: - перегрузка операторов с использованием функций-членов; - перегрузка унарных операторов; - перегрузка операторов с использованием функций-не членов класса; - перегрузка унарных операторов с использованием дружественных функций; - перегрузка оператора индексации массивов (<code>[]</code>); - перегрузка оператора <code>"()</code>".
11	Наследование Рассматриваемые вопросы: - понятие о наследовании; - управление доступом к членам базового класса; - использование защищенных членов класса; - спецификатор <code>protected</code> для наследования базового класса; - наследование нескольких базовых классов; - конструкторы, деструкторы и наследование; - передача параметров конструкторам базового класса; - предоставление доступа.
12	Виртуальные функции и полиморфизм Рассматриваемые вопросы: - определение полиморфизма классов объектов; - указатели на производные типы; - виртуальные функции; - наследование виртуальных функций; - чисто виртуальные функции и абстрактные классы; - раннее и позднее связывание.
13	Контейнер вектор Рассматриваемые вопросы: - вектор как динамический массив; - функции-члены класса <code>vector</code>; - использование итератора для доступа к вектору; - вставка и удаление элементов из вектора.
14	Контейнер списки Рассматриваемые вопросы: - функции-члены класса <code>list</code>; - сортировка списка; - объединение списков.

№ 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.

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

№ п/п	Наименование лабораторных работ / краткое содержание
17	С++ Конструкторы и деструкторы. В результате выполнения лабораторной работы студент получает навыки работы с разными типами конструкторов и деструкторами
18	С++ Конструкторы и деструкторы. 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 учебной литературой.
2	Работа с лекционным материалом.
3	Подготовка к лабораторным работам
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	Унгер, А. Ю. Объектно-ориентированное программирование : учебник / А. Ю. Унгер.	https://e.lanbook.com/book/398276 (дата обращения: 06.05.2026)

	— Москва : РТУ МИРЭА, 2023. — 102 с. — ISBN 978-5-7339-2051-1	
2	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	https://reader.lanbook.com/book/108512 (дата обращения: 19.06.2025)
3	Ашарина, И. В. Объектно-ориентированное программирование в C++: лекции и упражнения : учебное пособие / И. В. Ашарина. — 2-е изд., перераб. и доп. — Москва : Горячая линия-Телеком, 2017. — 336 с. — ISBN 978-5-9912-0423-1. — Текст : электронный // Лань : электронно-библиотечная система.	https://e.lanbook.com/book/119830 (дата обращения: 19.06.2025)
4	Страуструп, Б. Дизайн и эволюция C++ / Б. Страуструп. — Москва : ДМК Пресс, 2007. — 448 с. — ISBN 5-94074-005-7.	https://e.lanbook.com/book/1222 (дата обращения: 23.10.2025).

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);
- Образовательная платформа «Юрайт» (<https://urait.ru/>);
- Электронно-библиотечная система издательства «Лань» (<http://e.lanbook.com/>);

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:

и.о. заведующего кафедрой ММСС

С.А. Тищенко

Заведующий кафедрой ЦТУТП

В.Е. Нутович

Председатель учебно-методической
комиссии

Н.А. Андриянова