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

Задачами данной дисциплины является формирование у студента базовых знаний в области программирования на языке C++, а также 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:

- синтаксис, особенности программирования и стандартные библиотеки выбранного языка программирования;
- основные типы данных и базовые языковые конструкции;
- принципы объектно-ориентированного программирования.

Уметь:

- создавать блок-схемы алгоритмов функционирования разрабатываемых программных продуктов;
- осуществлять отладку программных продуктов;
- применять принципы объектно-ориентированного программирования с помощью выбранного языка программирования;
- использовать стандартные библиотеки для написания программного обеспечения.

Владеть:

- навыками изучения технической документации по языку программирования;

- навыками применения информации из технической документации по языку программирования при написании программного обеспечения;

- навыками составления блок-схемы алгоритма разрабатываемого программного обеспечения в соответствии с техническим заданием;

- навыками написания исходного кода программного обеспечения;

- навыками отладки разработанного кода программного обеспечения.

3. Объем дисциплины (модуля).

3.1. Общая трудоемкость дисциплины (модуля).

Общая трудоемкость дисциплины (модуля) составляет 10 з.е. (360 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	
		№1	№2
Контактная работа при проведении учебных занятий (всего):	160	96	64
В том числе:			
Занятия лекционного типа	64	32	32
Занятия семинарского типа	96	64	32

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 200 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	Препроцессорные средства Рассматриваемые вопросы: - директивы препроцессора; - замены в тексте; - условная компиляция.
3	Основы программирования Рассматриваемые вопросы: - лексика языка; - переменные и константы; - типы данных; - выражения и операции; - логические конструкции и ветвления; - конструкция if/else; - вложенные условия; - конструкция switch/case; - циклы ; - цикл for; - цикл while; - цикл do while; - вложенные циклы.
4	Процедурное программирование Рассматриваемые вопросы: - потоковый ввод/вывод; - массивы; - работа с массивами; - массивы и указатели; - символьная информация и строки; - общие сведения о функциях; - ссылки, перегрузка; - структуры; - ввод/вывод файлов.
5	Модели памяти и пространство имен Рассматриваемые вопросы: - раздельная компиляция; - автоматический класс памяти; - статический класс памяти; - спецификаторы классов памяти; - понятие пространства имен; - пространство имен std;
6	Объектно-ориентированное программирование Рассматриваемые вопросы: - введение в классы;

№ п/п	Тематика лекционных занятий / краткое содержание
	- доступ к членам класса; - конструкторы и деструкторы; - дружественные структуры; - инкапсуляция; - наследование; - виртуальные функции и полиморфизм.
7	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.
8	Введение в классы Рассматриваемые вопросы: - объекты и классы; - доступ к членам класса; - конструкторы и деструкторы; - типы конструкторов; - перегрузка конструкторов; - конструкторы, деструкторы и передача объектов; - возвращение объектов функциями; - сравнение структур с классами; - работа с указателем this; - дружественные функции; - встраиваемые функции; - массивы объектов; - указатели на объекты.
9	Инкапсуляция Рассматриваемые вопросы: - определение инкапсуляции классов объектов; - время жизни и область видимости; - локальность и глобальность; - приватная область; - защищенная область; - общедоступная область.
10	Доступ к членам класса Рассматриваемые вопросы: - общедоступная область; - приватная область; - защищённая область.
11	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.

№ п/п	Тематика лекционных занятий / краткое содержание
12	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 ([]); - перегрузка оператора "()".
13	Наследование Рассматриваемые вопросы: - понятие о наследовании; - управление доступом к членам базового класса; - использование защищенных членов класса; - спецификатор protected для наследования базового класса; - наследование нескольких базовых классов; - конструкторы, деструкторы и наследование; - передача параметров конструкторам базового класса; - предоставление доступа.
14	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.
15	Шаблоны Рассматриваемые вопросы: - шаблоны функций; - выведение типов шаблонных аргументов; - шаблоны классов; - спецификации; - шаблонные аргументы-константы; - передача шаблонных аргументов; - классы vector, list, queue, deque, stack.
16	Обработка исключительных ситуаций Рассматриваемые вопросы: - основы обработки исключительных ситуаций; - функции exit() и abort(); - перехват исключений классового типа; - перехват исключений базового класса; - исключение bad_alloc и оператор new.
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; - файловый ввод-вывод; - чтение и запись текстовых файлов;

№ п/п	Тематика лекционных занятий / краткое содержание
	- функции get() и put(); - работа в режиме командной строки.
18	Препроцессор C++ Рассматриваемые вопросы: - директива; - макроопределения, действующие как функции; - директива #include; - директивы условной компиляции; - операторы препроцессора "#" и "##".

4.2. Занятия семинарского типа.

Лабораторные работы

№ п/п	Наименование лабораторных работ / краткое содержание
1	C++ Основные типы данных и конструкции языка В результате выполнения лабораторных работ студент получает навыки работы с базовыми конструкциями, типами данных и синтаксическими особенностями языка C++
2	C++ Циклы В результате выполнения лабораторных работ студент получает навыки работы с операторами циклов (while, do while, for) на языке C++.
3	C++ Массивы В результате выполнения лабораторных работ студент получает навыки работы с массивами на языке C++.
4	C++ Строки В результате выполнения лабораторных работ студент получает навыки работы со строками и основными библиотечными функциями обработки строк на языке C++
5	C++ Функции В результате выполнения лабораторных работ студент получает навыки работы по созданию функций и передачи параметров различными способами на языке C++
6	C++ Структуры В результате выполнения лабораторных работ студент получает навыки работы со структурами.
7	C++ ООП В результате выполнения лабораторных работ студент получает навыки разработки алгоритмов программ с применением объектно-ориентированной подхода на языке C++, применяя теоретические знания в области принципов ООП (абстракция, наследование, полиморфизм, инкапсуляция).
8	C++ Конструкторы и деструкторы В результате выполнения лабораторной работы студент получает навыки описания конструкторов и деструкторов в классе объектов, приобретает понятие о перегрузки методов, о типах конструкторов и виртуализации деструкторов.
9	C++ Наследование В результате выполнения лабораторной работы студент получает навыки наследование одного класса от другого.
10	C++ Полиморфизм В результате выполнения лабораторной работы студент получает навыки переопределения методов в наследуемых классах.

№ п/п	Наименование лабораторных работ / краткое содержание
11	Обработка исключений В результате выполнения лабораторных работ студент получает навыки разработки алгоритмов программ с использованием технологий обработки исключений на языке C++.
12	C++ Стандартная библиотека шаблонов В результате выполнения лабораторных работ студент получает навыки работы со стандартной библиотекой шаблонов.
13	Потоковый ввод - вывод C++ тВ результате выполнения лабораторных работ студент получает навыки работы со стандартными потоками ввода-вывода.
14	Потоковый ввод - вывод C++ В результате выполнения лабораторных работ студент получает навыки работы со строковыми потоками ввода-вывода.
15	Потоковый ввод - вывод C++ В результате выполнения лабораторных работ студент получает навыки работы с файловыми потоками ввода-вывода.
16	Указатели в C++ В результате выполнения лабораторных работ студент получает навыки работы с именованными и неименованными областями памяти с помощью указателей.
17	Указатели в C++ В результате выполнения лабораторных работ студент получает навыки работы по передачи указателей в качестве параметров в функции.

Практические занятия

№ п/п	Тематика практических занятий/краткое содержание
1	Препроцессор C++ В результате выполнения практических работ студент получает навыки работы с директивами препроцессора.
2	Потоковый ввод-вывод C++ В результате выполнения практических работ студент получает навыки работы со стандартными потоками ввода-вывода.
3	Потоковый ввод-вывод C++ В результате выполнения практических работ студент получает навыки работы со строковыми потоками ввода-вывода.
4	Потоковый ввод-вывод C++ В результате выполнения практических работ студент получает навыки работы с файловыми потоками ввода-вывода.
5	Указатели в C++ В результате выполнения практических работ студент получает навыки работы с именованными и неименованными областями памяти с помощью указателей.
6	Указатели в C++ В результате выполнения практических работ студент получает навыки работы по передачи указателей в качестве параметров в функции.
7	Обработка исключений В результате выполнения практических работ студент получает навыки разработки алгоритмов программ с использованием технологий обработки исключений на языке C++
8	Стандартная библиотека шаблонов В результате выполнения практических работ студент получает навыки работы со стандартной библиотекой шаблонов.

4.3. Самостоятельная работа обучающихся.

№ п/п	Вид самостоятельной работы
1	Изучение рекомендованной литературы
2	Подготовка к лабораторным работам
3	Подготовка к практическим 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	Барков, И. А. Объектно-ориентированное программирование / И. А. Барков. — 2-е изд., стер. — Санкт-Петербург : Лань, 2023. — 700 с. — ISBN 978-5-507-47113-3.	https://e.lanbook.com/book/329549 (дата обращения: 30.01.2024).
2	Зоткин, С. П. Программирование на языке высокого уровня C/C++ : учебное пособие / С. П. Зоткин. — 3-е изд. — Москва : МИСИ – МГСУ, 2018. — 140 с. — ISBN 978-5-7264-1810-0.	https://e.lanbook.com/book/108512
3	Программирование на языке C++ : методические указания / С. В. Борисов, О. Б. Пащенко, И. Л. Серебрякова, В. П. Степанов. — Москва : МГТУ им. Н.Э. Баумана, 2017. — 76 с. — ISBN 978-5-7038-4726-8	https://e.lanbook.com/book/103498
4	Павловская, Т. А. Программирование на языке C++ : учебное пособие / Т. А. Павловская. — 2-е изд. — Москва : ИНТУИТ, 2016. — 154 с. — Текст : электронный // Лань : электронно-библиотечная система.	https://e.lanbook.com/book/100409
5	Кетков, Ю. Л. Введение в языки программирования C и C++ : учебное пособие / Ю. Л. Кетков. — 2-е изд. — Москва : ИНТУИТ, 2016. — 291 с. — Текст : электронный // Лань : электронно-библиотечная система.	https://e.lanbook.com/book/100719
6	Быков, А. Ю. Решение задач на языках программирования Си и Си++ : методические указания / А. Ю. Быков. — Москва : МГТУ им. Баумана, 2017. — 248 с. — ISBN 978-5-7038-4577-6	https://e.lanbook.com/book/103505

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

Электронно-библиотечная система Научно-технической библиотеки МИИТ (http://library.miit.ru/)

C/C++ tutorial (http://www.cplusplus.com/doc/tutorial).

C/C++ practice (https://www.geeksforgeeks.org/c-plus-plus).

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

Microsoft Visual Studio.

Microsoft Internet Explorer (или другой браузер).

Операционная система Microsoft Windows.

Microsoft Office.

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

Аудитория для проведения занятий лекционного типа должна быть оснащена персональным компьютером и набором демонстрационного оборудования.

Аудитория для проведения практических занятий должна быть оснащена персональными компьютерами.

9. Форма промежуточной аттестации:

Зачет в 1 семестре.

Экзамен во 2 семестре.

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.Клычева