

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



Рабочая программа дисциплины (модуля),  
как компонент образовательной программы  
базового высшего образования  
по направлению подготовки  
02.03.02 Фундаментальная информатика и  
информационные технологии,  
утвержденной первым проректором РУТ (МИИТ)  
Тимониным В.С.

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)**

**Алгоритмизация и программирование**

Направление подготовки: 02.03.02 Фундаментальная информатика и  
информационные технологии

Направленность (профиль): Квантовые вычислительные системы и сети

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: 4196  
Подписал: заведующий кафедрой Желенков Борис  
Владимирович  
Дата: 04.06.2026

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

Целью освоения учебной дисциплины «Алгоритмизация и программирование» является формирование у студентов целостных представлений о принципах алгоритмизации и программирования на примере языка C++.

Основными задачами дисциплины являются:

- изучение основ алгоритмизации;
- изучение языка C++;
- изучение средств разработки и отладки программ на языке C++;
- практика программирования на языке C++.

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

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

**ОПК-3** - Способен применять компьютерные/суперкомпьютерные методы, современное программное обеспечение, в том числе отечественного происхождения, для решения задач профессиональной деятельности;

**ОПК-4** - Способен к разработке алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, созданию информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям;

**ОПК-7** - Способен понимать принципы работы современных информационных технологий и использовать их для решения задач профессиональной деятельности.

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

### **Знать:**

- Основы информатики.
- Основные структуры данных и методы их обработки.
- Различия между языками программирования высокого и низкого уровня.
- Язык программирования Си.
- Набор функций стандартных библиотек.

### **Уметь:**

- Формализовать прикладную задачу, выбирать для неё подходящие структуры данных и алгоритмы обработки.

- Программировать алгоритмы, используя средства языков высокого уровня.

- Разрабатывать тестовые случаи и сценарии.

#### **Владеть:**

- Навыками разработки программ для ЭВМ на языке Си.

- Навыками в проведении отладки и тестирования разработанных программ.

- Навыками анализа получаемых результатов и оформлении документации на программу.

### 3. Объем дисциплины (модуля).

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

Общая трудоемкость дисциплины (модуля) составляет 8 з.е. (288 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	80	80
В том числе:			
Занятия лекционного типа	64	32	32
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 128 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	1 семестр . Введение:основные понятия и термины.Основные этапы решения задач Рассматриваемые вопросы: - основные понятия и термины программирования; - этапы решения задачи, для последующего написания программы.
2	Алгоритм Рассматриваемые вопросы: - понятие алгоритма, свойства, присущие алгоритму, способы описания алгоритма; - основные виды алгоритмов и способы их задания; - линейные алгоритмы; - разветвляющиеся алгоритмы; - циклические алгоритмы.
3	Характерные приемы алгоритмизации задач Рассматриваемые вопросы: - традиционные методы решения классических задач: запоминание результатов, вычисление суммы и произведения, вычисление наибольшего и наименьшего.
4	Массивы Рассматриваемые вопросы: - понятие массива, способы задания массивов; - одномерные массивы; - двумерные массивы.
5	Алгоритмы сортировки Рассматриваемые вопросы: - понятие сортировки, основные методы и алгоритмы сортировки; - алгоритм сортировки: «Метод пузырька»; - алгоритм сортировки «Метод попарных сравнений».
6	Обработка символьной информации Рассматриваемые вопросы: - понятие символьной информации; - способы обработки символов; - алгоритмы работы с символьной информацией.
7	Краткая характеристика языка C++ Рассматриваемые вопросы: - краткая характеристика языка C++; - примеры программ; - базовые средства языка C++; - состав языка. Идентификаторы. Ключевые слова. Знаки операций. Константы. Комментарии. Переменные и выражения. Операции.
8	Типы данных C++ Рассматриваемые вопросы:

№ п/п	Тематика лекционных занятий / краткое содержание
	<ul style="list-style-type: none"> <li>- концепция типов данных;</li> <li>- основные типы данных.</li> </ul>
9	<p><b>Базовые структуры структурного программирования</b></p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> <li>- понятие структурного программирования;</li> <li>- оператор «выражение»;</li> <li>- операторы ветвления. Примеры программ.</li> </ul>
10	<p><b>Операторы цикла</b></p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> <li>- реализация циклов в языке C++;</li> <li>- Простые циклы;</li> <li>- Вложенные циклы. Примеры программ.</li> </ul>
11	<p><b>Массивы</b></p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> <li>- реализация массивов в языке C++;</li> <li>- Одномерные массивы. Примеры программ;</li> <li>- Двумерные массивы. Примеры программ.</li> </ul>
12	<p><b>Указатели</b></p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> <li>- понятие указателя;</li> <li>- адресная арифметика;</li> <li>- связь между массивами и указателями;</li> <li>- бестиповый указатель;</li> <li>- ссылки. Примеры программ.</li> </ul>
13	<p><b>Обработка символьной информации</b></p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> <li>- способы обработки символьной информации в языке C++;</li> <li>- простейшие алгоритмы сортировки. «Метод пузырька». «Метод попарных сравнений». Примеры программ.</li> </ul>
14	<p><b>Функции</b></p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> <li>- реализация функций в языке;</li> <li>- объявление и определение функции;</li> <li>- глобальные и локальные переменные. Возвращаемое значение;</li> <li>- параметры функции. Рекурсивные функции;</li> <li>- перегрузка функций. Шаблоны функций. Функция main(). Функции стандартной библиотеки. Примеры программ.</li> </ul>
15	<p><b>Директивы процессора</b></p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> <li>- директива #include. Директива #define. Директивы условной компиляции. Директива #undef;</li> <li>- области действия идентификаторов. Внешние объявления; Поименованные области.</li> </ul>
16	<p><b>Ввод-вывод данных</b></p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> <li>- управление файловыми потоками. Открытие потоков;</li> <li>- перенаправление ввода – вывода. Управление буфером потока. Закрытие потоков;</li> <li>- ввод-вывод низкого уровня. Ввод-вывод символов.</li> </ul>
17	<p><b>2 семестр. Типы данных, определяемые пользователем</b></p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> <li>- Структуры. Массивы структур.</li> <li>- Битовые поля.</li> </ul>

№ п/п	Тематика лекционных занятий / краткое содержание
	<ul style="list-style-type: none"> <li>- Объединения.</li> <li>- Перечисления.</li> <li>- Примеры программ.</li> </ul>
18	<b>Классы</b> Рассматриваемые вопросы: <ul style="list-style-type: none"> <li>- Описание классов.</li> <li>- Описание объектов.</li> <li>- Указатель this.</li> </ul>
19	<b>Конструкторы</b> Рассматриваемые вопросы: <ul style="list-style-type: none"> <li>- Копирующий конструктор.</li> </ul>
20	<b>Конструкторы(продолжение)</b> Рассматриваемые вопросы: <ul style="list-style-type: none"> <li>- Статические элементы класса. Статические поля. Статические методы.</li> </ul>
21	<b>Конструкторы(продолжение)</b> Рассматриваемые вопросы: <ul style="list-style-type: none"> <li>- Дружественные конструкторы и классы.</li> </ul>
22	<b>Деструкторы и перегрузка</b> Рассматриваемые вопросы: <ul style="list-style-type: none"> <li>- Деструкторы</li> </ul>
23	<b>Деструкторы и перегрузка(продолжение)</b> Рассматриваемые вопросы: <ul style="list-style-type: none"> <li>- Перегрузка операций.</li> <li>- Перегрузка унарных операций.</li> <li>- Перегрузка бинарных операций.</li> </ul>
24	<b>Деструкторы и перегрузка(продолжение)</b> Рассматриваемые вопросы: <ul style="list-style-type: none"> <li>- Перегрузка операций присваивания.</li> <li>- Перегрузка операций new и delete.</li> </ul>
25	<b>Деструкторы и перегрузка(продолжение)</b> Рассматриваемые вопросы: <ul style="list-style-type: none"> <li>- Перегрузка операций приведения типа.</li> <li>- Перегрузка операций вызова функций.</li> <li>- Перегрузка операций индексирования.</li> </ul>
26	<b>Множественное наследование</b> Рассматриваемые вопросы: <ul style="list-style-type: none"> <li>- Множественное наследование</li> <li>- Отличие структур и объединений от классов.</li> </ul>
27	<b>Шаблоны классов</b> Рассматриваемые вопросы: <ul style="list-style-type: none"> <li>- Создание шаблона класса.</li> <li>- Использование шаблонов классов.</li> </ul>
28	<b>Шаблоны классов(продолжение)</b> Рассматриваемые вопросы: <ul style="list-style-type: none"> <li>- Специализация шаблона класса.</li> <li>- Достоинства и недостатки шаблонов.</li> </ul>
29	<b>Технология создания программ</b> Рассматриваемые вопросы:

№ п/п	Тематика лекционных занятий / краткое содержание
	- Кодирования и документирования программы. - Проектирование и тестирование программы.
30	Динамические структуры данных Рассматриваемые вопросы: - Линейные списки. - Стеки.
31	Динамические структуры данных(продолжение) Рассматриваемые вопросы: - Очереди. - Бинарные деревья.
32	Динамические структуры данных(продолжение) Рассматриваемые вопросы: - Реализация динамических структур с помощью массивов.

## 4.2. Занятия семинарского типа.

### Лабораторные работы

№ п/п	Наименование лабораторных работ / краткое содержание
1	Семестр 1 Линейные алгоритмы  В результате выполнения лабораторной работы студент приобретает навык разработки и реализации линейных алгоритмов.
2	Разветвляющиеся алгоритмы В результате выполнения лабораторной работы студент получает навык разработки и реализации разветвляющихся алгоритмов.
3	Циклические алгоритмы В результате выполнения лабораторной работы студент приобретает навык разработки и реализации циклов с предусловием
4	Циклические алгоритмы(продолжение) В результате выполнения лабораторной работы студент приобретает навык разработки и реализации циклов с постусловием.
5	Циклические алгоритмы(продолжение) В результате выполнения лабораторной работы студент приобретает навык разработки и реализации циклов типа for
6	Итерационный вычислительный процесс В результате выполнения лабораторной работы студент учится выделять закономерности в различных вычислительных процессах и разрабатывать итерационные алгоритмы для реализации алгоритмов поиска суммы и произведения ряда.
7	Итерационный вычислительный процесс(продолжение) В результате выполнения лабораторной работы студент учится выделять закономерности в различных вычислительных процессах и разрабатывать итерационные алгоритмы для реализации алгоритмов математических функций, рядов.
8	Одномерные массивы В результате выполнения лабораторной работы студент получает навык разработки и реализации алгоритмов для обработки одномерных массивов.

№ п/п	Наименование лабораторных работ / краткое содержание
9	<b>Обработка символьной информации</b> В результате выполнения лабораторной работы студент приобретает навык разработки и реализации алгоритмов для обработки символьной информации.
10	<b>Двумерные массивы</b> В результате выполнения лабораторной работы студент получает навык разработки и реализации алгоритмов для обработки двумерных целочисленных массивов.
11	<b>Двумерные массивы(продолжение)</b> В результате выполнения лабораторной работы студент получает навык разработки и реализации алгоритмов для обработки двумерных символьных массивов.
12	<b>Сортировка данных</b> В результате выполнения лабораторной работы студент приобретает навык разработки и реализации алгоритма сортировки «Пузырьком».
13	<b>Сортировка данных(продолжение)</b> В результате выполнения лабораторной работы студент приобретает навык разработки и реализации алгоритма сортировки «Попарные сравнения».
14	<b>Обработка строковых данных</b> В результате выполнения лабораторной работы студент приобретает навык разработки и реализации алгоритмов ввода/вывода строк.
15	<b>Обработка строковых данных(продолжение)</b> В результате выполнения лабораторной работы студент приобретает навык разработки и реализации алгоритмов работы со строками типа char.d
16	<b>Обработка строковых данных(продолжение)</b> В результате выполнения лабораторной работы студент приобретает навык разработки и реализации алгоритмов работы со строками типа string.
17	<b>2 семестр</b> Обработка символьной информации с использованием самостоятельно разработанных функций  В результате выполнения лабораторной работы студент учится разрабатывать функции для обработки символьной информации, работающие с глобальными переменными.
18	<b>Обработка символьной информации с использованием самостоятельно разработанных функций(продолжение)</b> В результате выполнения лабораторной работы студент учится разрабатывать функции для обработки символьной информации, работающие с формальными параметрами и локальными переменными.
19	<b>Разработка функций для решения математических задач</b> В результате выполнения лабораторной работы студент отрабатывает умения разработки собственных функций для решения бесконечно возрастающей последовательности с использованием формальных параметров.
20	<b>Разработка функций для решения математических задач(продолжение)</b> В результате выполнения лабораторной работы студент отрабатывает умения разработки собственных функций для поиска суммы ряда ограниченной последовательности с использованием глобальных переменных.
21	<b>Разработка функций для обработки двумерных массивов</b> В результате выполнения лабораторной работы студент отрабатывает навык создания функций и обработки двумерных массивов для поиска максимума и минимума.

№ п/п	Наименование лабораторных работ / краткое содержание
22	Разработка функций для обработки двумерных массивов(продолжение) В результате выполнения лабораторной работы студент отрабатывает навык создания функций и обработки двумерных массивов для сортировки строк и столбцов двумерного массива.
23	Работа с файлами и разработка соответствующих функций В результате выполнения лабораторной работы студент получает навык по работе с файлами и разработке и реализации собственных функций для отображения данных из файла и записи введенных данных в файл.
24	Работа с файлами и разработка соответствующих функций(продолжение) В результате выполнения лабораторной работы студент получает навык по работе с файлами и разработке и реализации собственных функций для обработки информации, полученной из файла.
25	Создание и обработка наборов объектов заданной структуры В результате выполнения лабораторной работы студент приобретает навык разработки и реализации структуры класса.
26	Создание и обработка наборов объектов заданной структуры В результате выполнения лабораторной работы студент приобретает навык разработки методов ввода-вывода.
27	Создание и обработка наборов объектов заданной структуры(продолжение) В результате выполнения лабораторной работы студент приобретает навык разработки методов обработки и взаимодействия объектов.
28	Создание и обработка наборов данных заданной структуры с использованием конструкторов В результате выполнения лабораторной работы студент получает навык реализации конструктора по умолчанию.
29	Создание и обработка наборов данных заданной структуры с использованием конструкторов(продолжение) В результате выполнения лабораторной работы студент получает навык разработки и реализации конструктора с параметрами.
30	Создание и обработка наборов данных заданной структуры с использованием конструкторов(продолжение) В результате выполнения лабораторной работы студент получает навык разработки и реализации конструктора с параметрами по умолчанию.
31	Создание и обработка наборов объектов заданной структуры с использованием статических членов-данных и членов-функций Создание и обработка наборов объектов заданной структуры с использованием статических членов-данных и членов-функций
32	Создание и обработка наборов данных заданной структуры с использованием перегрузки встроенных операций В результате выполнения лабораторной работы студент получает навык реализации и применения перегрузки встроенных операций.
33	Создание и обработка наборов данных заданной структуры с использованием перегрузки операторов ввода-вывода В результате выполнения лабораторной работы студент получает навык реализации операторов ввода.
34	Создание и обработка наборов данных заданной структуры с использованием перегрузки операторов ввода-вывода(продолжение)

№ п/п	Наименование лабораторных работ / краткое содержание
	В результате выполнения лабораторной работы студент получает навык реализации операторов вывода.
35	Создание и обработка наборов объектов заданной структуры с использованием динамической памяти В результате выполнения лабораторной работы студент получает навык использования динамической памяти при реализации программ модификации существующих методов для использования с оператором new
36	Создание и обработка наборов объектов заданной структуры с использованием динамической памяти(продолжение) В результате выполнения лабораторной работы студент получает навык использования динамической памяти при реализации программ модификации существующих методов для использования с оператором delete.
37	Создание и обработка наборов объектов заданной структуры с использованием динамической памяти(продолжение) В результате выполнения лабораторной работы студент получает навык использования динамической памяти при реализации деструктора для класса.
38	Создание и обработка наборов объектов заданной структуры с использованием возможностей наследования классов. В результате выполнения лабораторной работы студент получает навык использования наследования классов при реализации программ дочернего класса, иллюстрирующего наследование public.
39	Создание и обработка наборов объектов заданной структуры с использованием возможностей наследования классов(продолжение) В результате выполнения лабораторной работы студент получает навык использования наследования классов при реализации программ дочернего класса, иллюстрирующего наследование private.
40	Создание и обработка наборов объектов заданной структуры с использованием возможностей наследования классов(продолжение) В результате выполнения лабораторной работы студент получает навык использования наследования классов при реализации программ дочернего класса, иллюстрирующего наследование protected.

### Практические занятия

№ п/п	Тематика практических занятий/краткое содержание
1	1 семестр Линейные алгоритмы В результате выполнения практического задания студент отрабатывает умения разработки линейных и разветвляющихся алгоритмов.
2	Циклические алгоритмы В результате выполнения практического задания студент отрабатывает умения разработки циклических алгоритмов различных видов.
3	Одномерные массивы В результате выполнения практического задания студент отрабатывает умения разработки алгоритмов обработки одномерных массивов.

№ п/п	Тематика практических занятий/краткое содержание
4	<b>Двумерные массивы</b> В результате выполнения практического задания студент отработывает умения разработки алгоритмов обработки двумерных массивов.
5	<b>Обработка символьной информации</b> В результате выполнения практического задания студент отработывает умения разработки алгоритмов для обработки символьной информации.
6	<b>Сортировка</b> В результате выполнения практического задания студент отработывает умения разработки алгоритмов сортировки.
7	<b>Написание программ</b> В результате выполнения практического задания студент отработывает навык написания программ по разработанным ранее алгоритмам.
8	<b>Работа с файлами и разработка соответствующих функций</b> В результате выполнения практического задания студент получает навык по работе с файлами и написанию соответствующих программных решений.

#### 4.3. Самостоятельная работа обучающихся.

№ п/п	Вид самостоятельной работы
1	Изучение дополнительной литературы
2	Подготовка к практическим занятиям
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	А.В. Михайлюк, Введение в объектно-ориентированное программирование [Текст] : учеб. пособие по дисц. "Программирование на языке высокого уровня" и "Технологии программирования" для студ. напр. "Информатика и выч. техника", МИИТ. Каф. "Вычислительные системы и сети". - М. : МИИТ, 2009. - 340 с.	<a href="http://library.miit.ru/miitpublishing/10-1525.pdf">http://library.miit.ru/miitpublishing/10-1525.pdf</a>
2	В.Н. Нагинаев, Основы алгоритмизации и программирования на языке С++ [Текст] : Учебное пособие. - М. : МИИТ, 2009. - 204 с	<a href="http://library.miit.ru/miitpublishing/10-1284.pdf">http://library.miit.ru/miitpublishing/10-1284.pdf</a>

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

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

Форум специалистов по информационным технологиям  
<http://citforum.ru/>

Интернет-университет информационных технологий  
<http://www.intuit.ru/>

Тематический форум по информационным технологиям  
<http://habrahabr.ru/>

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

Microsoft Windows

Microsoft Office

Интернет-браузер (Yandex и др.)

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

Учебная аудитория для проведения учебных занятий (занятий лекционного типа, практических занятий, лабораторных работ):

- компьютер преподавателя, мультимедийное оборудование, рабочие станции студентов, доска.

Аудитория подключена к сети «Интернет».

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  
«Вычислительные системы и  
квантовые коммуникации»

Н.А. Цыганова

Согласовано:

Заведующий кафедрой ВССиИБ

Б.В. Желенков

Председатель учебно-методической  
комиссии

Н.А. Андриянова