

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
10.03.01 Информационная безопасность,
утвержденной первым проректором РУТ (МИИТ)
Тимониным В.С.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

Алгоритмизация и программирование

Направление подготовки: 10.03.01 Информационная безопасность

Направленность (профиль): Безопасность компьютерных систем

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: 30.01.2026

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-2 - Способен применять информационно-коммуникационные технологии, программные средства системного и прикладного назначения, в том числе отечественного производства, для решения задач профессиональной деятельности ;

ОПК-7 - Способен использовать языки программирования и технологии разработки программных средств 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	<p>1 СЕМЕСТР Введение:основные понятия и термины.Основные этапы решения задач</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - основные понятия и термины программирования; - этапы решения задачи, для последующего написания программы.
2	<p>Алгоритм</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - понятие алгоритма, свойства, присущие алгоритму, способы описания алгоритма; - основные виды алгоритмов и способы их задания; - линейные алгоритмы; - разветвляющиеся алгоритмы; - циклические алгоритмы.
3	<p>Характерные приемы алгоритмизации задач</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - традиционные методы решения классических задач: запоминание результатов, вычисление суммы и произведения, вычисление наибольшего и наименьшего.
4	<p>Массивы</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - понятие массива, способы задания массивов; - одномерные массивы; - двумерные массивы.
5	<p>Алгоритмы сортировки</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - понятие сортировки, основные методы и алгоритмы сортировки; - алгоритм сортировки: «Метод пузырька»; - алгоритм сортировки «Метод попарных сравнений».
6	<p>Обработка символьной информации</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - понятие символьной информации; - способы обработки символов; - алгоритмы работы с символьной информацией
7	<p>Краткая характеристика языка C++</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - краткая характеристика языка C++; - примеры программ; - базовые средства языка C++; - состав языка. Идентификаторы. Ключевые слова. Знаки операций. Константы. Комментарии. Переменные и выражения. Операции
8	<p>Типы данных C++</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - концепция типов данных; - основные типы данных.
9	<p>Базовые структуры структурного программирования</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - понятие структурного программирования;

№ п/п	Тематика лекционных занятий / краткое содержание
	<ul style="list-style-type: none"> - Оператор «выражение»; - Операторы ветвления. Примеры программ.
10	Операторы цикла Рассматриваемые вопросы: <ul style="list-style-type: none"> - реализация циклов в языке C++; - Простые циклы; - Вложенные циклы. Примеры программ.
11	Массивы Рассматриваемые вопросы: <ul style="list-style-type: none"> - реализация массивов в языке C++; - Одномерные массивы. Примеры программ; - Двумерные массивы. Примеры программ.
12	Указатели Рассматриваемые вопросы: <ul style="list-style-type: none"> - понятие указателя; - Адресная арифметика; - Связь между массивами и указателями; - Бестиповый указатель; - Ссылки. Примеры программ.
13	Обработка символьной информации Рассматриваемые вопросы: <ul style="list-style-type: none"> - способы обработки символьной информации в языке C++; - Простейшие алгоритмы сортировки. «Метод пузырька». «Метод попарных сравнений». Примеры программ.
14	Функции Рассматриваемые вопросы: <ul style="list-style-type: none"> - реализация функций в языке; - Объявление и определение функции; - Глобальные и локальные переменные. Возвращаемое значение; - Параметры функции. Рекурсивные функции; - Перегрузка функций. Шаблоны функций. Функция main(). Функции стандартной библиотеки. Примеры программ.
15	Директивы процессора Рассматриваемые вопросы: <ul style="list-style-type: none"> - Директива #include. Директива #define. Директивы условной компиляции. Директива #undef; - Области действия идентификаторов. Внешние объявления. Поименованные области.
16	Ввод-вывод данных Рассматриваемые вопросы: <ul style="list-style-type: none"> - Управление файловыми потоками. Открытие потоков; - Перенаправление ввода – вывода. Управление буфером потока. Закрытие потоков; - Ввод-вывод низкого уровня. Ввод-вывод символов.
17	2 СЕМЕСТР Типы данных, определяемые пользователем Рассматриваемые вопросы: <ul style="list-style-type: none"> - Структуры. Массивы структур. - Битовые поля. - Объединения. - Перечисления. - Примеры программ.

№ п/п	Тематика лекционных занятий / краткое содержание
18	Классы Рассматриваемые вопросы: - Описание классов. - Описание объектов. - Указатель this.
19	Конструкторы Рассматриваемые вопросы: - Копирующий конструктор
20	Конструкторы(продолжение) Рассматриваемые вопросы: - Статические элементы класса. Статические поля. Статические методы.
21	Конструкторы(продолжение) Рассматриваемые вопросы: - Дружественные конструкторы и классы.
22	Деструкторы и перегрузка - Деструкторы
23	Деструкторы и перегрузка(продолжение) Рассматриваемые вопросы: - Перегрузка операций. Перегрузка унарных операций. Перегрузка бинарных операций.
24	Деструкторы и перегрузка(продолжение) Рассматриваемые вопросы: - Перегрузка операций присваивания. - Перегрузка операций new и delete.
25	Деструкторы и перегрузка(продолжение) Рассматриваемые вопросы: - Перегрузка операций приведения типа. - Перегрузка операций вызова функций. - Перегрузка операций индексирования.
26	Множественное наследование Рассматриваемые вопросы: - Множественное наследование - Отличие структур и объединений от классов.
27	Шаблоны классов Рассматриваемые вопросы: - Создание шаблона класса. - Использование шаблонов классов.
28	Шаблоны классов(продолжение) Рассматриваемые вопросы: - Специализация шаблона класса. - Достоинства и недостатки шаблонов.
29	Технология создания программ 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	1 СЕМЕСТР Лабораторная работа №1. (2 часа) Линейные алгоритмы В результате выполнения лабораторной работы студент приобретает навык разработки и реализации линейных алгоритмов.
2	Лабораторная работа №2. (2 часа) Разветвляющиеся алгоритмы В результате выполнения лабораторной работы студент получает навык разработки и реализации разветвляющихся алгоритмов.
3	Лабораторная работа №3-5 (6 часов). Циклические алгоритмы – циклы с предусловием (2 часа), циклы с постусловием (2 часа), циклы типа for (2 часа) В результате выполнения лабораторной работы студент приобретает навык разработки и реализации различных видов циклических алгоритмов.
4	Лабораторная работа №6-7(4 часа). Итерационный вычислительный процесс – алгоритмы поиска суммы и произведения ряда (2 часа), реализация алгоритмов математических функций, рядов (2 часа) В результате выполнения лабораторной работы студент учится выделять закономерности в различных вычислительных процессах и разрабатывать итерационные алгоритмы для реализации
5	Лабораторная работа №8. (2 часа) Одномерные массивы В результате выполнения лабораторной работы студент получает навык разработки и реализации алгоритмов для обработки одномерных массивов.
6	Лабораторная работа №9. (2 часа) Обработка символьной информации

№ п/п	Наименование лабораторных работ / краткое содержание
	В результате выполнения лабораторной работы студент приобретает навык разработки и реализации алгоритмов для обработки символьной информации.
7	Лабораторная работа №10-11 (4 часа). Двумерные массивы – работа с двумерными целочисленными массивами (2 часа), работа с символьными двумерными массивами (2 часа) В результате выполнения лабораторной работы студент получает навык разработки и реализации алгоритмов для обработки двумерных массивов.
8	Лабораторная работа №12-13 (4 часа). Сортировка данных – алгоритм сортировки «Пузырьком» (2 часа), алгоритм сортировки «Попарные сравнения» (2 часа) В результате выполнения лабораторной работы студент приобретает навык разработки и реализации различных видов алгоритмов сортировки данных.
9	Лабораторная работа №14-16. (6 часов) Обработка строковых данных – алгоритмы ввода/вывода строк (2 часа), работа со строками типа char (2 часа), работа со строками типа string (2 часа) В результате выполнения лабораторной работы студент приобретает навык разработки и реализации алгоритмов для обработки строк.
10	2 СЕМЕСТР Лабораторная работа № 1-2. (4 часа) Обработка символьной информации с использованием самостоятельно разработанных функций – разработка и реализация функций, работающих с глобальными переменными (2 часа), разработка и реализация функций, работающих с формальными параметрами и локальными переменными (2 часа). В результате выполнения лабораторной работы студент учится разрабатывать функции для обработки символьной информации в соответствии с поставленной задачей
11	Лабораторная работа № 3-4. (4 часа) Разработка функций для решения математических задач – разработка собственных функций для решения бесконечно возрастающей последовательности с использованием формальных параметров (2 часа), разработка собственных функций для поиска суммы ряда ограниченной последовательности с использованием глобальных переменных (2 часа). В результате выполнения лабораторной работы студент отрабатывает умения разработки функции на примере решения математических задач.
12	Лабораторная работа №5-6. (4 часа) Разработка функций для обработки двумерных массивов – разработка собственных функций для поиска максимума и минимума в двумерном массиве (2 часа), разработка собственных функций для сортировки строк и столбцов двумерного массива (2 часа) В результате выполнения лабораторной работы студент отрабатывает навык создания функций и обработки двумерных массивов.
13	Лабораторная работа № 7-8. (4 часа) Работа с файлами и разработка соответствующих функций – разработка и реализация собственных функций для отображения данных из файла и записи введенных данных в файл (2 часа), разработка и реализация собственных функций для обработки информации, полученной из файла (2 часа) . В результате выполнения лабораторной работы студент получает навык по работе с файлами и написанию соответствующих программных решений.
14	Лабораторная работа №9-11. (6 часов) Создание и обработка наборов объектов заданной структуры – разработка и реализация структуры класса и прототипов методов (2 часа), разработка методов ввода-вывода (2 часа), разработка методов обработки и взаимодействия объектов (2 часа) В результате выполнения лабораторной работы студент приобретает навык разработки и реализации классов и экземпляров класса (объектов).

№ п/п	Наименование лабораторных работ / краткое содержание
15	<p>Лабораторная работа № 12-14. (6 часов)</p> <p>Создание и обработка наборов данных заданной структуры с использованием конструкторов – разработка и реализация конструктора по умолчанию (2 часа), разработка и реализация конструктора с параметрами (2 часа), разработка и реализация конструктора с параметрами по умолчанию (2 часа)</p> <p>В результате выполнения лабораторной работы студент получает навык реализации конструкторов для создания классов и объектов класса.</p>
16	<p>Лабораторная работа № 15. (2 часа)</p> <p>Создание и обработка наборов объектов заданной структуры с использованием статических членов-данных и членов-функций.</p> <p>В результате выполнения лабораторной работы студент приобретает навык реализации и использования статических полей класса.</p>
17	<p>Лабораторная работа № 16. (2 часа)</p> <p>Создание и обработка наборов данных заданной структуры с использованием перегрузки встроенных операций</p> <p>В результате выполнения лабораторной работы студент получает навык реализации и применения перегрузки встроенных операций.</p>
18	<p>Лабораторная работа № 17-18. (4 часа)</p> <p>Создание и обработка наборов данных заданной структуры с использованием перегрузки операторов ввода-вывода – реализация перегрузки оператора ввода (2 часа), реализация перегрузки оператора вывода (2 часа)</p> <p>В результате выполнения лабораторной работы студент получает навык реализации и применения перегрузки операторов ввода-вывода.</p>
19	<p>Лабораторная работа № 19-21. (6 часов)</p> <p>Создание и обработка наборов объектов заданной структуры с использованием динамической памяти – модификация существующих методов для использования с оператором new (2 часа), модификация существующих методов для использования с оператором delete (2 часа), реализация деструктора для класса (2 часа)</p> <p>В результате выполнения лабораторной работы студент получает навык использования динамической памяти при реализации программ.</p>
20	<p>Лабораторная работа № 22-24. (6 часов)</p> <p>Создание и обработка наборов объектов заданной структуры с использованием возможностей наследования классов – разработка дочернего класса, иллюстрирующего наследование public (2 часа), разработка дочернего класса, иллюстрирующего наследование private (2 часа), разработка дочернего класса, иллюстрирующего наследование protected (2 часа),</p> <p>В результате выполнения лабораторной работы студент получает навык использования наследования классов при реализации программ.</p>

Практические занятия

№ п/п	Тематика практических занятий/краткое содержание
1	<p>1 СЕМЕСТР Линейные и разветвляющиеся алгоритмы</p> <p>В результате выполнения практического задания студент отрабатывает умения разработки линейных и разветвляющихся алгоритмов.</p>
2	<p>. Циклические алгоритмы</p> <p>В результате выполнения практического задания студент отрабатывает умения разработки циклических алгоритмов различных видов.</p>

№ п/п	Тематика практических занятий/краткое содержание
3	Одномерные массивы В результате выполнения практического задания студент отрабатывает умения разработки алгоритмов обработки одномерных массивов.
4	Двумерные массивы В результате выполнения практического задания студент отрабатывает умения разработки алгоритмов обработки двумерных массивов.
5	Обработка символьной информации В результате выполнения практического задания студент отрабатывает умения разработки алгоритмов для обработки символьной информации.
6	Сортировка В результате выполнения практического задания студент отрабатывает умения разработки алгоритмов сортировки.
7	Написание программ В результате выполнения практического задания студент отрабатывает навык написания программ по разработанным ранее алгоритмам.
8	Работа с файлами и разработка соответствующих функций 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	Т.А. Павловская , С/С++. Программирование на языке высокого уровня [Текст] : учебник для вузов /. - ISBN 5-94723-568-4, СПб. : Питер, 2007. - 461 с. : ил. - ("Учебник для вузов").	РУТ(МИИТ), Фундаментальная библиотека (ауд. 1230), полочный шифр 004 П12(дата обращения 31.03.2025)
2	В.Н. Нагинаев, Основы алгоритмизации и программирования на языке С++ [Текст] : Учебное пособие. - М. : МИИТ, 2009. - 204 с	http://library.miit.ru/miitpublishing/10-1284.pdf (дата обращения 31.03.2025) Текст электронный

3	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.	http://library.miit.ru/miitpublishing/10-1525.pdf (дата обращения 31.03.2025) Текст электронный
---	--	---

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

- 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, курсового проектирования (выполнения курсовых работ), текущего контроля и промежуточной аттестации):

- компьютер преподавателя, проектор, экран проекционный, рабочие станции студентов, маркерная доска.

Аудитория подключена к сети «Интернет»

9. Форма промежуточной аттестации:

Экзамен в 1, 2 семестрах.

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, сети и
информационная безопасность»

Н.А. Цыганова

Согласовано:

Заведующий кафедрой ВССиИБ

Б.В. Желенков

Председатель учебно-методической
комиссии

Н.А. Андриянова