

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
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: 24.10.2024

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 компьютерные/суперкомпьютерные методы, современное программное обеспечение, в том числе отечественного происхождения, для решения задач профессиональной деятельности;

ОПК-3 - 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-4 - Способен участвовать в разработке технической документации программных продуктов и комплексов с использованием стандартов, норм и правил, а также в управлении проектами создания информационных систем на стадиях жизненного цикла.

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++

№ п/п	Тематика лекционных занятий / краткое содержание
	<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; - оператор «выражение»; - операторы ветвления. Примеры программ.
10	<p>Операторы цикла</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - реализация циклов в языке C++; - Простые циклы; - Вложенные циклы. Примеры программ.
11	<p>Массивы</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - реализация массивов в языке C++; - Одномерные массивы. Примеры программ; - Двумерные массивы. Примеры программ.
12	<p>Указатели</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - понятие указателя; - адресная арифметика; - связь между массивами и указателями; - бестиповый указатель; - ссылки. Примеры программ.
13	<p>Обработка символьной информации</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - способы обработки символьной информации в языке C++; - простейшие алгоритмы сортировки. «Метод пузырька». «Метод попарных сравнений». Примеры программ.
14	<p>Функции</p> <p>Рассматриваемые вопросы:</p> <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	<p>Директивы процессора</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - директива #include. Директива #define. Директивы условной компиляции. Директива #undef; - области действия идентификаторов. Внешние объявления. Поименованные области.
16	<p>Ввод-вывод данных</p> <p>Рассматриваемые вопросы:</p> <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	<p>2 семестр. Типы данных, определяемые пользователем</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - Структуры. Массивы структур.

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

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

№ п/п	Наименование лабораторных работ / краткое содержание
	обработки двумерных массивов для сортировки строк и столбцов двумерного массива.
23	<p>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.</p>
24	<p>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.</p>
25	<p>Создание и обработка наборов объектов заданной структуры В результате выполнения лабораторной работы студент приобретает навык разработки и реализации структуры класса.</p>
26	<p>Создание и обработка наборов объектов заданной структуры В результате выполнения лабораторной работы студент приобретает навык разработки методов ввода-вывода.</p>
27	<p>Создание и обработка наборов объектов заданной структуры(продолжение) В результате выполнения лабораторной работы студент приобретает навык разработки методов обработки и взаимодействия объектов.</p>
28	<p>Создание и обработка наборов данных заданной структуры с использованием конструкторов В результате выполнения лабораторной работы студент получает навык реализации конструктора по умолчанию.</p>
29	<p>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.</p>
30	<p>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.</p>
31	<p>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</p>
32	<p>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.</p>
33	<p>Создание и обработка наборов данных заданной структуры с использованием перегрузки операторов ввода-вывода В результате выполнения лабораторной работы студент получает навык реализации операторов ввода.</p>
34	<p>Создание и обработка наборов данных заданной структуры с использованием перегрузки операторов ввода-вывода(продолжение) В результате выполнения лабораторной работы студент получает навык реализации операторов вывода.</p>
35	<p>Создание и обработка наборов объектов заданной структуры с использованием динамической памяти</p>

№ п/п	Наименование лабораторных работ / краткое содержание
	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.
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.

№ п/п	Тематика практических занятий/краткое содержание
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	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
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

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:

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