

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
09.03.01 Информатика и вычислительная техника,
утвержденной первым проректором РУТ (МИИТ)
Тимониным В.С.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

Алгоритмы и структуры данных

Направление подготовки: 09.03.01 Информатика и вычислительная техника

Направленность (профиль): Технологии разработки программного обеспечения

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: 5665
Подписал: заведующий кафедрой Нутович Вероника
Евгеньевна
Дата: 01.09.2024

1. Общие сведения о дисциплине (модуле).

Целями освоения данной дисциплины являются получение базовых, теоретических знаний в области структур данных, их реализация на языках высокого уровня.

Основной задачей изучения дисциплины является формирование базовых представлений, знаний и умений в области алгоритмов работы структур данных, преимущества и недостатки различных структур данных, пользовательской реализации как структур данных, так и алгоритмов поиска.

2. Планируемые результаты обучения по дисциплине (модулю).

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

ПК-1 - Способен разрабатывать программные продукты используя разные языки программирования для корпоративного рынка.;

ПК-8 - Способен разрабатывать программные продукты используя современные методологии и практики для корпоративного рынка.

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

Уметь:

- применять структуры данных для решения профессиональных задач. Выбирать оптимальные структуры данных под конкретные задачи;
- применять методы динамического программирования для решения профессиональных задач;
- применять принципы объектно-ориентированного программирования для реализации структур данных.

Знать:

- линейные структуры данных, их особенности реализации, преимущества и недостатки;
- древовидные структуры данных, их особенности реализации, преимущества и недостатки;
- алгоритмы сортировки, их особенности реализации, преимущества и недостатки.
- основные методы определения вычислимой сложности алгоритмов.

Владеть:

- навыками реализации линейных структур данных на языках высокого уровня;

- навыками реализации древовидных структур данных на языках высокого уровня;
- навыками реализации алгоритмов с использованием рекурсии и бектрекинга;
- навыками определения временной и емкостной сложности алгоритмов.

3. Объем дисциплины (модуля).

3.1. Общая трудоемкость дисциплины (модуля).

Общая трудоемкость дисциплины (модуля) составляет 3 з.е. (108 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 №5
Контактная работа при проведении учебных занятий (всего):	48	48
В том числе:		
Занятия лекционного типа	16	16
Занятия семинарского типа	32	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 60 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>Введение в абстрактные типы данных.</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - память и хранение данных в ЭВМ. - абстрактные структуры данных; - алгоритмическая сложность.
2	<p>Линейные структуры данных. Массивы и списки. Стек и очередь.</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - динамические массивы; - связанные списки (односвязные, двусвязные, циклические) - операции, преимущества и недостатки; - реализация на основе массивов и списков; - применение стеков (вызов функций, вычисление выражений) и очередей (моделирование, планировщики).
3	<p>Древовидные структуры данных. Бинарные деревья. Бинарное дерево поиска (BST). Сбалансированные деревья поиска.</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - терминология (корень, узел, лист, высота, глубина); - обходы деревьев (прямой, симметричный, обратный, в ширину); - бинарные деревья - свойства, представление; - операции (поиск, вставка, удаление); - эффективность операций; - проблема сбалансированности; - AVL-деревья, Красно-черные деревья - основные идеи балансировки, сложность операций; - B-деревья и их применение.
4	<p>Куча. Хэш-таблицы.</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - бинарная куча (min-heap, max-heap) - свойства, операции (вставка, извлечение минимума/максимума, построение кучи); - применение куч (пирамидальная сортировка); - хэш-функции; - методы разрешения коллизий (цепочки, открытая адресация); - эффективность операций; - применение хэш-таблиц.
5	<p>Алгоритмы сортировки. Алгоритмы поиска.</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - простые сортировки (пузырьковая, выбором, вставками); - эффективные сортировки - быстрая сортировка (Quicksort), сортировка слиянием (Mergesort), пирамидальная сортировка (Heapsort); - сортировки, не основанные на сравнениях; - линейный поиск; - бинарный поиск; - интерполяционный поиск.
6	<p>Графы. Рекурсия и бэктрекинг. Жадные алгоритмы.</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - основные понятия (вершины, ребра, ориентированные/неориентированные, взвешенные графы); - представление графов (матрица смежности, списки смежности); - алгоритмы обхода графов (DFS, BFS); - кратчайшие пути (Дейкстра, Беллман-Форд);

№ п/п	Тематика лекционных занятий / краткое содержание
	<ul style="list-style-type: none"> - понятие рекурсии, примеры рекурсивных алгоритмов; - метод «разделяй и властвуй»; - бэктрекинг - общая схема, применение для решения переборных задач (задача о N ферзях); - принцип жадного выбора; - примеры задач, решаемых жадными алгоритмами (задача о выборе заявок, задача о рюкзаке с непрерывными весами).
7	<p>Динамическое программирование</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - принцип оптимальности Беллмана; - примеры задач - числа Фибоначчи, задача о наибольшей общей подпоследовательности, задача о рюкзаке.
8	<p>Комбинаторные алгоритмы. Строковые алгоритмы.</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - генерация перестановок, сочетаний, размещений; - поиск подстроки (наивный, Кнут-Моррис-Пратт, Бойер-Мур); - сжатие данных.

4.2. Занятия семинарского типа.

Практические занятия

№ п/п	Тематика практических занятий/краткое содержание
1	<p>Анализ алгоритмов и O-нотация</p> <p>В результате работы на практическом занятии студент получает навык анализа временной и пространственной сложности кода. Студент учится определять асимптотическую сложность для простых циклов, вложенных циклов и рекурсивных вызовов, а также сравнивать эффективность различных алгоритмических подходов на практике.</p>
2	<p>Реализация линейных структур данных на основе массивов. Стек и очередь</p> <p>В результате работы на практическом занятии студент получает навык программной реализации стека и очереди на базе динамического массива. Студент осваивает операции push/pop для стека и enqueue/dequeue для очереди, а также решает практические задачи (например, проверка сбалансированности скобок, вычисление выражений в обратной польской нотации).</p>
3	<p>Реализация связанных списков</p> <p>В результате работы на практическом занятии студент получает навык программной реализации односвязных и двусвязных списков. Студент разрабатывает алгоритмы для основных операций: вставка, удаление, поиск элемента и обход списка, а также анализирует их сложность в сравнении с операциями над массивами.</p>
4	<p>Применение алгоритмов сортировки. Простые сортировки</p> <p>В результате работы на практическом занятии студент получает навык реализации и анализа простых алгоритмов сортировки (пузырьковая, выбором, вставками). Студент наглядно оценивает их квадратичную временную сложность и понимает области их ограниченного применения.</p>
5	<p>Применение алгоритмов сортировки. Эффективные сортировки</p> <p>В результате работы на практическом занятии студент получает навык реализации эффективных алгоритмов сортировки, основанных на парадигме «разделяй и властвуй»: быстрая сортировка (Quicksort) и сортировка слиянием (Mergesort)</p>

№ п/п	Тематика практических занятий/краткое содержание
6	Применение алгоритмов поиска В результате работы на практическом занятии студент получает навык реализации линейного и бинарного поиска. Студент на практике убеждается в преимуществе бинарного поиска для отсортированных данных и оценивает его логарифмическую сложность.
7	Рекурсивные алгоритмы В результате работы на практическом занятии студент получает навык решения задач с использованием рекурсии. Студент разрабатывает рекурсивные функции для классических задач (числа Фибоначчи, факториал, обход дерева) и учится понимать и отлаживать рекурсивный вызов.
8	Реализация бинарного дерева поиска (BST) В результате работы на практическом занятии студент получает навык работы с древовидными структурами данных. Студент реализует операции вставки, поиска, удаления узла, а также различные виды обхода дерева (прямой, симметричный, обратный).
9	Сбалансированные деревья поиска В результате работы на практическом занятии студент изучает проблему разбалансировки BST и знакомится с концепцией AVL-деревьев. Студент получает навык реализации или анализа алгоритмов малых и больших вращений для поддержания сбалансированности дерева.
10	Реализация двоичной кучи (Binary Heap) В результате работы на практическом занятии студент получает навык реализации структуры данных «приоритетная очередь» на основе двоичной кучи. Студент разрабатывает алгоритмы для операций вставки и извлечения элемента с максимальным/минимальным приоритетом и применяет их для реализации пирамидальной сортировки (Heapsort).
11	Реализация хэш-таблицы В результате работы на практическом занятии студент получает навык программной реализации хэш-таблиц. Студент разрабатывает хэш-функции и реализует один из методов разрешения коллизий (например, метод цепочек), оценивая эффективность операций вставки и поиска.
12	Представление графов и алгоритмы обхода В результате работы на практическом занятии студент получает навык представления графов в памяти (матрица смежности, списки смежности). Студент реализует фундаментальные алгоритмы обхода графов: поиск в глубину (DFS) и поиск в ширину (BFS).
13	Алгоритмы поиска кратчайшего пути в графе В результате работы на практическом занятии студент получает навык решения оптимизационных задач на графах. Студент реализует алгоритм Дейкстры для нахождения кратчайших путей от одной вершины до всех остальных во взвешенном графе.
14	Жадные алгоритмы В результате работы на практическом занятии студент осваивает принципы построения жадных алгоритмов. Студент решает классические задачи, такие как задача о выборе заявок или задача о рюкзаке с непрерывными весами, и анализирует условия применимости жадного подхода.
15	Основы динамического программирования В результате работы на практическом занятии студент получает базовый навык решения задач методом динамического программирования. На примере классических задач (например, нахождение наибольшей общей подпоследовательности, задача о рюкзаке) студент осваивает принципы построения рекуррентных соотношений и их реализации.
16	Алгоритмы на строках В результате работы на практическом занятии студент получает навык разработки и применения алгоритмов для обработки текстовых данных. Студент реализует наивный алгоритм поиска подстроки и один из эффективных алгоритмов, например, алгоритм Кнута-Морриса-Пратта (КМП), анализируя их производительность.

4.3. Самостоятельная работа обучающихся.

№ п/п	Вид самостоятельной работы
1	Работа с лекционным материалом.
2	Работа с литературой.
3	Текущая подготовка к 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	Тюкачев, Н. А. С#. Алгоритмы и структуры данных : учебное пособие для вузов / Н. А. Тюкачев, В. Г. Хлебостроев. — 4-е изд., стер. — Санкт-Петербург : Лань, 2021. — 232 с. — ISBN 978-5-8114-8247-4	https://e.lanbook.com/book/172708
2	Кораблин, Ю. П. Структуры и алгоритмы обработки данных : учебно-методическое пособие / Ю. П. Кораблин, В. П. Сыромятников, Л. А. Скворцова. — Москва : РТУ МИРЭА, 2020. — 219 с.	https://e.lanbook.com/book/163860
3	Пантелеев, Е. Р. Алгоритмы и структуры данных : учебное пособие / Е. Р. Пантелеев, А. Л. Алыкова. — Иваново : ИГЭУ, 2018. — 142 с.	https://e.lanbook.com/book/154576
4	Хиценко, В. П. Структуры данных и алгоритмы : учебное пособие / В. П. Хиценко. — Новосибирск : НГТУ, 2016. — 64 с. — ISBN 978-5-7782-2958-7.	https://e.lanbook.com/book/118222
5	Хиценко, В. П. Структуры данных и алгоритмы : учебное пособие / В. П. Хиценко. — Новосибирск : НГТУ, 2016. — 64 с. — ISBN 978-5-7782-2958-7	https://e.lanbook.com/book/118222

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

Q&A-портал программистов (<https://ru.stackoverflow.com/>)

Открытые лекции (<https://sphere.mail.ru/materials/video/#16>)

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).

Microsoft Internet Explorer (или другой браузер).

Операционная система Microsoft Windows.

Microsoft Office

.NET 6

Java 17

Microsoft Visual Studio CE

JetBrains IntelliJ IDEA Community Edition

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

Учебные аудитории для проведения учебных занятий, оснащенные компьютерной техникой и наборами демонстрационного оборудования.

Для практических занятий – наличие персональных компьютеров вычислительного класса.

9. Форма промежуточной аттестации:

Экзамен в 5 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»

Е.А. Заманов

Согласовано:

Заведующий кафедрой ЦТУТП

В.Е. Нутович

Председатель учебно-методической
комиссии

Н.А. Андриянова