

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«МУРМАНСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Кафедра математики,  
информационных систем  
и программного обеспечения

**МЕТОДИЧЕСКИЕ УКАЗАНИЯ  
К САМОСТОЯТЕЛЬНОЙ РАБОТЕ**

По дисциплине: Б1.О.09.03 Структуры и алгоритмы обработки данных  
название дисциплины

для направления подготовки 09.03.01 Информатика и вычислительная техника  
направленность (профиль) «Программное обеспечение вычислительной техники  
и автоматизированных систем»  
квалификация выпускника бакалавр

Мурманск  
2020

Составитель - Романовская Юлия Владимировна, канд. физ.-мат. наук, доцент кафедры математики, информационных систем и программного обеспечения

Методические указания к самостоятельной работе рассмотрены и одобрены на заседании кафедры-разработчика:

математики, информационных систем и программного обеспечения

«24» ноября 2020 г., протокол № 4.

## ОБЩИЕ МЕТОДИЧЕСКИЕ УКАЗАНИЯ

Данные методические указания разработаны в соответствии с ФГОС ВО по направлению подготовки

09.03.01 Информатика и вычислительная техника

код и наименование направления подготовки

утверждённого 19.09.2017 г. № 929 и учебного плана направления подготовки

(дата, номер приказа Минобрнауки РФ)

(обозначение или наименование документа университетского уровня)

09.03.01 Информатика и вычислительная техника (направленность программы:

Программное обеспечение вычислительной техники и автоматизированных систем)

Целью дисциплины «Структуры и алгоритмы обработки данных» является формирование компетенций в соответствии с ФГОС ВО по направлению подготовки 09.03.01 Информатика и вычислительная техника и учебным планом в составе ОПОП по направлению подготовки 09.03.01 Информатика и вычислительная техника, направленность (профиль): «Программное обеспечение вычислительной техники и автоматизированных систем», что предполагает формирование у обучающегося знаний о применяемых в программировании структурах данных, их спецификации и реализации, алгоритмах обработки данных, взаимосвязи алгоритмов и структур данных.

Задачи: дать необходимые знания по методам разработки эффективных алгоритмов и программ, созданию структур данных, анализа эффективности алгоритмов.

Процесс изучения дисциплины направлен на формирование общепрофессиональных компетенций в соответствии с ФГОС ВО по направлению подготовки 09.03.01 Информатика и вычислительная техника:

- способен разрабатывать алгоритмы и программы, пригодные для практического применения (ОПК-8).

В результате изучения дисциплины обучающийся должен:

**знать:**

- алгоритмические языки программирования, операционные системы и оболочки, современные среды разработки программного обеспечения;

**уметь:**

- составлять алгоритмы, писать и отлаживать коды на языке программирования, тестировать работоспособность программы, интегрировать программные модули;

**владеть:**

- языком программирования; навыками отладки и тестирования работоспособности программы.

## ТЕМАТИЧЕСКИЙ ПЛАН

Наименование тем и содержание самостоятельной работы	Количество часов по формам обучения	
	Очная	Заочная
<p><b>Линейные и нелинейные структуры данных (СД)</b></p> <p>1. СД: основные определения, абстрактный тип данных. Функциональная спецификация, логическое описание и физическое представление и реализация. Линейные структуры данных: стеки, очереди и деки.</p> <p>2. Кольцевые(циклические) линейные списки. Массивы. Представление разреженных массивов. Множества. Представление множества в виде линейного списка, массива и характеристического вектора.</p> <p>3. Графы и способы их машинного представления. Деревья и бинарные деревья; их машинное представление</p> <p>4. Пример использования бинарных деревьев в задаче упаковки сообщений: префиксные коды и бинарные деревья, метод кодирования Фано-Шеннона, метод кодирования (сжатия) информации по Хаффмену.</p>	10	32
<p><b>Методы разработки эффективных алгоритмов</b></p> <p>5. Алгоритм и его свойства. Временная и пространственная сложность алгоритма. Итерация и рекурсия; их достоинства и недостатки. Обходы деревьев: применение рекурсии для обхода бинарных деревьев</p> <p>6. Метод «разделяй и властвуй» на примере алгоритма быстрого умножения. Бинарный (дихотомический) поиск. Балансировка (уравновешивание) на примере метода дихотомического поиска.</p> <p>7. Динамическое программирование на примере алгоритмов для определения оптимального порядка перемножения матриц, вероятности победы в спортивных соревнованиях.</p> <p>8. Динамическое программирование на примере задачи триангуляции.</p>	6	20
<p><b>Рекуррентные уравнения</b></p> <p>9. Задача о ханойской башне. Задача о разрезании пиццы. Задача Иосифа Флавия. Некоторые рекуррентные уравнения и их решение.</p>	8	20
<p><b>Алгоритмы сортировки</b></p> <p>10. Задачи сортировки. Внутренняя и внешняя сортировки. Базовые идеи алгоритмов сортировки. Сортировка распределением: цифровая сортировка («вычерпыванием»).</p> <p>11. Анализ сложности и эффективности алгоритмов поиска и сортировки с помощью сравнений. Сортировка включением: сортировка простым включением, двоичным включением и метод Шелла. Временная сложность алгоритмов.</p> <p>12. Сортировка слиянием. Сортировка обменами: пузырьковая и оптимальная (быстрая) сортировка. Временная сложность алгоритмов.</p> <p>13. Сортировка извлечением: сортировка с помощью прямого выбора и древесная сортировка, сортировка лесом. Временная сложность алгоритмов.</p> <p>14. Топологическая сортировка. Временная сложность алгоритма.</p> <p>15. Внешняя сортировка: поглощением, двухпутевым челночным слиянием. Сортировка многопутевым челночным слиянием. Временная сложность алгоритмов.</p> <p>16. Внешняя сортировка разделительным методом ПАРОМСОРТ. Временная сложность алгоритма.</p>	10	30

<p><b>Быстрый поиск</b></p> <p>15. Задача поиска и кодирования (сжатия) данных. Поиск в таблице: последовательный и дихотомический. Дерево поиска.</p> <p>16. Идеально сбалансированное двоичное дерево. Сбалансированные по высоте (АВЛ) деревья поиска и их построение.</p> <p>17. Включение в сбалансированное двоичное дерево, исключение из сбалансированного двоичного дерева.</p> <p>18. Построение дерева оптимального поиска. Хорошие бинарные деревья поиска.</p> <p>19. Файлы: организация и обработка, представления деревьями: Б-деревья. Включение в Б-дерево, исключение из Б-дерева</p> <p>20. Метод поиска с использованием функции расстановки (хеширование). Разрешение коллизий: метод внутренних и внешних цепочек, метод открытой адресации.</p> <p>21. Задача поиска подстроки. Алгоритм Кнута-Мориса-Пратта. Алгоритм Боуера-Мура.</p>	12	40
<p><b>Исчерпывающий поиск</b></p> <p>22. Алгоритмы с возвратом (backtracking) и их свойства. Задача о ходе коня и алгоритм ее решения. Задача о восьми ферзях и алгоритм ее решения.</p> <p>23. Задача о стабильных браках и алгоритм решения.</p> <p>24. Задача оптимального выбора и алгоритм решения.</p>	8	12
<p><b>Метод ветвей и границ</b></p> <p>25. Метод ветвей и границ.</p> <p>26. Задача почтальона. Задача коммивояжера и её решение методом ветвей и границ.</p>	8	18
<p><b>Алгоритмы на графах</b></p> <p>27. Минимальное остовное дерево. Алгоритм нахождения эйлера цикла в графе. Алгоритм нахождения гамильтонова цикла в графе</p> <p>28. Схемы поиска в глубину и в ширину в графе. Нахождение компонент двусвязности: точки сочленения графа. Алгоритм нахождения компонент двусвязности.</p> <p>29. Паросочетания графов. Задача о наибольшем паросочетании.</p> <p>30. Клики. Алгоритм порождения клик графа.</p> <p>31. Кратчайшие пути в графе. Алгоритмы Форда-Беллмана и Дейкстры для нахождения расстояний от вершины-источника до остальных вершин.</p> <p>32. Расстояния в бесконтурном графе. Расстояния между всеми парами вершин (алгоритм Флойда-Уоршалла).</p> <p>33. Нахождение максимального потока в сети. Алгоритм Форда-Фалкерсона.</p> <p>34. Задачи размещения. Задачи поиска центра, задачи поиска медиан.</p> <p>35. Сетевые графики. Метод критического пути.</p>	8	25
<b>Итого:</b>	80	16

## СПИСОК РЕКОМЕНДУЕМОЙ ЛИТЕРАТУРЫ

### *Основная литература*

1. Царёв, Р.Ю. Алгоритмы и структуры данных (CDIO) / Р.Ю. Царёв, А.В. Прокопенко ; Министерство образования и науки Российской Федерации, Сибирский Федеральный университет. – Красноярск : СФУ, 2016. – 204 с. : ил. – Режим доступа: по подписке. – URL: <http://biblioclub.ru/index.php?page=book&id=497016>
2. Самуйлов, С. В. Алгоритмы и структуры обработки данных : учебное пособие / С. В. Самуйлов. — Саратов : Вузовское образование, 2016. — 132 с. — ISBN 2227-8397. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <http://www.iprbookshop.ru/47275.html>

3. Костюкова, Н. И. Комбинаторные алгоритмы для программистов / Н. И. Костюкова. — М. : Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. — 216 с. — ISBN 978-5-9556-0069-7. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <http://www.iprbookshop.ru/52192.html>

### **Дополнительная литература**

1. Теория алгоритмов / сост. А.А. Брыкалова ; Министерство образования и науки РФ, Федеральное государственное автономное образовательное учреждение высшего образования «Северо-Кавказский федеральный университет». — Ставрополь : СКФУ, 2016. — 129 с. : ил. — Режим доступа: по подписке. — URL: <http://biblioclub.ru/index.php?page=book&id=467402>
2. Алексеев, В. Е. Графы и алгоритмы. Структуры данных. Модели вычислений / В. Е. Алексеев, В. А. Таланов. — М. : Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. — 153 с. — ISBN 5-9556-0066-3. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <http://www.iprbookshop.ru/52186.html>
3. Быкова, В.В. Комбинаторные алгоритмы: множества, графы, коды / В.В. Быкова ; Министерство образования и науки Российской Федерации, Сибирский федеральный университет. — Красноярск : Сибирский федеральный университет, 2015. — 152 с. : табл., ил. — Режим доступа: по подписке. — URL: <http://biblioclub.ru/index.php?page=book&id=435666>
4. Комлева, Н. В. Структуры и алгоритмы компьютерной обработки данных : учебное пособие / Н. В. Комлева. — М. : Евразийский открытый институт, Московский государственный университет экономики, статистики и информатики, 2004. — 140 с. — ISBN 5-7764-0400-2. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <http://www.iprbookshop.ru/10898.html>

### **Перечень ресурсов информационно - телекоммуникационной сети «Интернет», необходимых для освоения дисциплины**

Электронно-библиотечная система IPR BOOKS (<http://www.iprbookshop.ru>)  
ЭБС «Университетская библиотека онлайн» (<http://biblioclub.ru/>)

Полнотекстовые базы данных, научные электронные библиотеки

1. Научная электронная библиотека (eLIBRARY.RU): <http://elibrary.ru>
2. ЭБД РГБ (Электронная библиотека диссертаций Российской Государственной библиотеки) <http://diss.rsl.ru/>
3. Реферативно-аналитическая база данных «Scopus» <http://www.scopus.com/home.uri>

### **МЕТОДИЧЕСКИЕ УКАЗАНИЯ К САМОСТОЯТЕЛЬНОЙ РАБОТЕ ПО ВЫПОЛНЕНИЮ РАСЧЕТНО-ГРАФИЧЕСКОЙ РАБОТЫ**

Внимательно прочитать задания расчетно-графической работы, повторить теоретический материал по конкретному заданию и приступить к выполнению работы. Готовую программу проверить на указанных тестовых наборах.

#### **РГР «Структуры данных и эффективные алгоритмы»**

##### **Задание 1. Работа с графами и деревьями**

Написать программу, которая проверяет, существует ли в неориентированном графе с 5 и более вершинами цикл, содержащий 4 заданные вершины в любом порядке следования. Граф представлен списками инцидентности и считывается из файла.

*Тестовый набор 1:*

Списки инцидентности:

1 → 2 → 3

2 → 1 → 3 → 4

3 → 1 → 2 → 4

4 → 2 → 3 → 5

5 → 4

Вершины: 1, 2, 3, 4

*Результат на тестовом наборе 1:*

Цикл, содержащий вершины 1, 2, 3, 4, существует

*Тестовый набор 2:*

Списки инцидентности:

1 → 2 → 3

2 → 1 → 3 → 4

3 → 1 → 2

4 → 2 → 5

5 → 4

Вершины: 1, 2, 3, 4

*Результат на тестовом наборе 2:*

Цикл, содержащий вершины 1, 2, 3, 4, не существует

**Задание 2.** Распознавание формулы и вычисление булевой функции

Формула, реализующая булеву функцию не более чем от 3-х переменных, задаётся (с клавиатуры или из текстового файла) символьным выражением с использованием следующих булевых операций:  $\vee$  - дизъюнкции и  $\&$  - конъюнкции. Написать программу, которая:

- 1) распознает символьную информацию и создаёт соответствующее данной формуле бинарное дерево в виде триплетов (вершина, левый сын, правый сын);
- 2) выдаёт на экран или в файл таблицу значений булевой функции;
- 3) проверяет, принадлежит ли данная функция одному из известных замкнутых классов.

Замкнутый класс, на принадлежность которому нужно проверить функцию, определяется путем нахождения остатка от деления номера студента по журналу на число 4:

- остаток 0 –  $T_0$  – класс функций, сохраняющий константу 0;
- остаток 1 –  $T_1$  – класс функций, сохраняющий константу 1;
- остаток 2 –  $S$  – класс самодвойственных функций;
- остаток 3 –  $M$  – класс монотонных функций.

*Тестовый набор 1:*

$x \vee y \& z$

Результат на тестовом наборе 1:

1) ( $\vee$ , x, &), (x, null, null), (&, y, z), (y, null, null), (z, null, null)

**Примечание:** порядок перечисления триплетов может быть отличен от указанного выше.

2)

x	y	z	$x \vee y \& z$
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

3) Функция принадлежит классу  $T_0$

Функция принадлежит классу  $T_1$

Функция не принадлежит классу S

Функция принадлежит классу M

**Примечание:** класс, подлежащий проверке, зависит от варианта задания

Тестовый набор 2:

$x \& y \vee y \& z$

Результат на тестовом наборе 1:

4) ( $\vee$ , &, &), (&, x, y), (&, y, z), (x, null, null), (y, null, null), (y, null, null), (z, null, null)

**Примечание:** порядок перечисления триплетов может быть отличен от указанного выше.

5)

x	y	z	$x \& y \vee y \& z$
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

6) Функция принадлежит классу  $T_0$

Функция принадлежит классу  $T_1$

Функция не принадлежит классу S

Функция принадлежит классу M

**Примечание:** класс, подлежащий проверке, зависит от варианта задания

### Задание 3. Алгоритмы внутренней сортировки

Написать программу, которая сортирует числовую последовательность случайных чисел из N элементов одним из методов сортировки. Методом дихотомического поиска

найти заданный элемент и вывести на экран его порядковый номер в упорядоченной последовательности.

Метод сортировки, который нужно применить, определяется путем нахождения остатка от деления номера студента по журналу на число 3:

- остаток 0 – метод Шелла;
- остаток 1 – метод быстрой сортировки;
- остаток 2 – метод древесной сортировки.

Тестовые наборы генерируются случайным образом с количеством элементов 100, 1000, 10000.

#### **Задание 4. Работа с деревом поиска**

Во входном текстовом файле содержатся исходные данные в виде записей с ключом. Организовать записи в виде дерева поиска. В качестве ключа использовать значения одного из полей. Перестроить дерево поиска, производя последовательно следующие операции:

- добавив новую запись,
- удалив запись из корня.

Реализовать функцию просмотра структуры дерева путем выдачи на экран его представления в виде триплетов (содержимое вершины, содержимое левого потомка, содержимое правого потомка).

Структура записей и ключевое поле определяются путем нахождения остатка от деления номера студента по журналу на число 3:

- остаток 0 – Фамилия И.О., рост, вес (ключевое поле **рост**);
- остаток 1 – Фамилия И.О., должность, стаж (ключевое поле **стаж**);
- остаток 2 – название организации, уставной капитал, год регистрации (ключевое поле **уставной капитал**).

Тестовые наборы определяются вариантом задания и содержат минимум 10 записей.