

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное бюджетное образовательное учреждение  
высшего профессионального образования  
«РЯЗАНСКИЙ ГОСУДАРСТВЕННЫЙ РАДИОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ  
имени В.Ф. Уткина»

Кафедра «Электронные вычислительные машины»

**ОЦЕНОЧНЫЕ МАТЕРИАЛЫ**  
**ПО ДИСЦИПЛИНЕ**  
**«Структуры и алгоритмы компьютерной обработки данных»**

Специальность  
02.03.03 «Математическое обеспечение  
и администрирование информационных систем»

Специализация  
Программное обеспечение компьютерных технологий и систем искусственного интеллекта

Квалификация выпускника – *бакалавр*  
Форма обучения – очная

Рязань

## **1 ОБЩИЕ ПОЛОЖЕНИЯ**

Оценочные материалы – это совокупность учебно-методических материалов (практических заданий, описаний форм и процедур проверки), предназначенных для оценки качества освоения обучающимися данной дисциплины как части ОПОП.

Цель – оценить соответствие знаний, умений и владений, приобретенных обучающимся в процессе изучения дисциплины, целям и требованиям ОПОП в ходе проведения промежуточной аттестации.

Основная задача – обеспечить оценку уровня сформированности компетенций, закрепленных за дисциплиной.

Контроль знаний обучающихся проводится в форме промежуточной аттестации.

Промежуточная аттестация проводится в форме экзамена. Форма проведения экзамена - тестирование, письменный опрос по теоретическим вопросам.

## **2 ОПИСАНИЕ ПОКАЗАТЕЛЕЙ И КРИТЕРИЕВ ОЦЕНИВАНИЯ КОМПЕТЕНЦИЙ**

Сформированность каждой компетенции (или ее части) в рамках освоения данной дисциплины оценивается по трехуровневой шкале:

- 1) пороговый уровень является обязательным для всех обучающихся по завершении освоения дисциплины;
- 2) продвинутый уровень характеризуется превышением минимальных характеристик сформированности компетенций по завершении освоения дисциплины;
- 3) эталонный уровень характеризуется максимально возможной выраженной степенью компетенций и является важным качественным ориентиром для самосовершенствования.

**Уровень освоения компетенций, формируемых дисциплиной:**

**Описание критериев и шкалы оценивания тестирования:**

<b>Шкала оценивания</b>	<b>Критерий</b>
3 балла (эталонный уровень)	уровень усвоения материала, предусмотренного программой: процент верных ответов на тестовые вопросы от 85 до 100%
2 балла (продвинутый уровень)	уровень усвоения материала, предусмотренного программой: процент верных ответов на тестовые вопросы от 70 до 84%
1 балл (пороговый уровень)	уровень усвоения материала, предусмотренного программой: процент верных ответов на тестовые вопросы от 50 до 69%
0 баллов	уровень усвоения материала, предусмотренного программой: процент верных ответов на тестовые вопросы от 0 до 49%

**Описание критериев и шкалы оценивания теоретического вопроса:**

<b>Шкала оценивания</b>	<b>Критерий</b>
3 балла (эталонный уровень)	выставляется студенту, который дал полный ответ на вопрос, показал глубокие систематизированные знания, смог привести примеры, ответил на дополнительные вопросы преподавателя
2 балла (продвинутый уровень)	выставляется студенту, который дал полный ответ на вопрос, но на некоторые дополнительные вопросы преподавателя ответил только с помощью наводящих вопросов
1 балл (пороговый уровень)	выставляется студенту, который дал неполный ответ на вопрос в билете и смог ответить на дополнительные вопросы только с помощью преподавателя
0 баллов	выставляется студенту, который не смог ответить на вопрос

**На промежуточную аттестацию (экзамен для ОФ или зачет для ВФ) выносится тест из 10-ми или 15-ти вопросов, два теоретических вопроса, две практические задачи.** Максимально студент может набрать 9 баллов. Итоговый суммарный балл студента, полученный при прохождении промежуточной аттестации, переводится в традиционную форму по системе «отлично», «хорошо», «удовлетворительно» и «неудовлетворительно».

**Оценка «отлично»** выставляется студенту, который набрал в сумме 9 баллов (выполнил все задания на эталонном уровне). Обязательным условием является выполнение всех предусмотренных в течение семестра практических заданий.

**Оценка «хорошо»** выставляется студенту, который набрал в сумме от 7 до 8 баллов при условии выполнения всех заданий на уровне не ниже продвинутого. Обязательным условием является выполнение всех предусмотренных в течение семестра практических заданий.

**Оценка «удовлетворительно»** выставляется студенту, который набрал в сумме от 5 до 6 баллов при условии выполнения всех заданий на уровне не ниже порогового. Обязательным условием является выполнение всех предусмотренных в течение семестра практических заданий.

**Оценка «неудовлетворительно»** выставляется студенту, который набрал в сумме менее 5 баллов или не выполнил всех предусмотренных в течение семестра практических заданий.

### 3 Паспорт фонда оценочных материалов по дисциплине

№ п/п	Контролируемые темы дисциплины	Код контролируемой компетенции (или её части)	Вид, форма оценочного мероприятия
1	Введение	ОПК-2	экзамен
2	Статические структуры данных (СД)	ОПК-2	Отчет, по ПЗ экзамен, зачет.
3	Линейно-динамические СД	ОПК-2	Отчет, по ПЗ экзамен, зачет
4	Полустатические СД	ОПК-2	Отчет, по ПЗ экзамен, зачет
5	Нелинейные связные СД	ОПК-2	Отчет, по ПЗ экзамен, зачет
6	Алгоритмы обработки данных	ОПК-2, ОПК-3	Отчет, по ПЗ экзамен, зачет

### 4 ТИПОВЫЕ КОНТРОЛЬНЫЕ ЗАДАНИЯ ИЛИ ИНЫЕ МАТЕРИАЛЫ

#### Промежуточная аттестация в форме экзамена (зачета)

##### Код компетенции ОПК-2

##### Типовые тестовые вопросы

№	Вопрос	Варианты ответов	
1	Структурой данных называется	1	Множество различных элементарных данных.
		+2	Информационный объект, состоящий из множества элементов данных, рассматриваемых и

			обрабатываемых совместно с помощью специальных операций.
		3	Совокупность правил и ограничений, отражающих отношения между элементами данных.
2	На логическом уровне данные представляются как	1	Объекты, существующие в терминах представлений и понятий пользователя.
		+2	Информационные объекты, предназначенные для выражения фактов и идей в формализованном виде с целью их обработки с помощью некоторого алгоритма или процесса.
		3	Информационные объекты, которые используются в программе пользователя.
3	Статическими называются структуры данных, в которых	1	Число и элементов, и связей между ними могут быть неизменными
		+2	Число и элементов, и связей между ними всегда постоянны
		3	Связи между элементами постоянны, но их количество может меняться
4	Дескриптором структуры данных называется	1	Дополнительная информация о физических параметрах структуры данных.
		2	Запись, в полях которой представляется различная детальная информация о рассматриваемой структуре данных.
		+3	Внутренняя структура, содержащая информацию об основных физических параметрах данной структуры данных.
5	Преобразование много мерной логической структуры массива в физическую может быть выполнено	1	Путем отображения строк в линейную последовательность ячеек оперативной памяти
		+2	Путем выполнения процесса линеаризации
		3	Путем отображения строк или столбцов в одномерную физическую структуру данных.
6	Связным списком называется	1	Последовательность записей одного и того же формата, связанных друг с другом.
		+2	Динамическая структура данных, элементами которой являются записи одинакового формата, связанные друг с другом с помощью одного или нескольких указателей, хранящихся в самих элементах
		3	Множество записей одинакового формата, связанных друг с другом с помощью одного или нескольких указателей.
7	Стеком называется	1	Последовательный список с переменной длиной, изменение числа элементов в котором всегда выполняется через вершину стека.

		2	Полустатическая структура, в которой всегда обрабатывается первым тот элемент, который был включен последним.
--	--	---	---

		+3	Полустатическая структура данных, включение и исключение элементов из которой выполняется только с одной стороны – вершины стека.
8	Чтобы получить кольцевую очередь необходимо	1	Конец очереди соединить с ее началом.
		+2	Перевести указатель, адресующий первый свободный элемент после последнего занятого, на первый элемент.
		3	Новый элемент всегда записывать по адресу первого элемента
9	Деревом называется	1	Нелинейная динамическая структура данных, каждый элемент которой может быть связан с другими элементами произвольными указателями.
		+2	Сетевая структура данных, в которой имеется единственный элемент структуры (узел) на которой не ссылается никакой другой элемент и который называется корнем дерева
		3	Нелинейная динамическая структура данных, в которой, начиная от корня и следуя по цепочке указателей можно получить доступ к любому элементу структуры.
10	Бинарным называется дерево, в котором	1	степень исхода для каждого узла равна 0.
		2	степень исхода для каждого узла может принимать значения 1 или 2
		+3	степень исхода для каждого узла не может превышать значение 2.
11	Высотой дерева называется	1	число ветвей от корня дерева до любого из конечных узлов.
		+2	максимальная длина пути, измеряемая в количестве ветвей от корня дерева до одного из листьев.
		3	длина пути от корня дерева до наиболее удаленного узла.
12	AVL-деревом называется	1	полностью сбалансированное двоичное дерево.
		2	двоичное дерево поиска, в котором левые и правые поддеревья любого узла имеют одинаковую высоту.

		+3	двоичное дерево поиска, в котором высоты левых и правых поддеревьев любого узла отличаются не более чем на 1, и в котором левые и правые поддеревья также являются АВЛ-деревьями.
--	--	----	---

#### 4.1.1.2. Типовые тестовые вопросы

1. Понятие и общее описание структур данных (СД). Уровни представления СД.
2. Классификация СД. Операции над СД.
3. Статические СД. Векторы.
4. Массивы.
5. Записи и таблицы.
6. Линейные СД. Понятие списка.
7. Линейно-динамические СД. Одно- и двухсвязные списки.
8. Основные операции на связных списках.
9. Полустатические СД. Стеки и их применение в вычислительных системах.
10. Очереди и их применение в вычислительных системах.
11. Нелинейные связные СД. Сетевые и древовидные СД. Основные понятия.
12. Бинарные деревья. Представление бинарных деревьев в ОП.

#### 4.1.1.3 Типовые практические задачи

1. Составить БСА и написать фрагмент программы (на языке Object Pascal или C++) для создания структуры односвязного списка из 8-ми элементов вещественного типа.
2. Составить БСА и написать фрагмент программы (на языке Object Pascal или C++) для создания структуры двусвязного списка из 10-ми элементов целого типа.
3. Составить БСА и написать фрагмент программы (на языке Object Pascal или C++) для создания структуры двусвязного списка из 10-ми элементов целого типа
4. Составить БСА и написать фрагмент программы (на языке Object Pascal или C++) для создания структуры очереди из 12-ми элементов целого типа.
5. Составить БСА и написать фрагмент программы (на языке Object Pascal или C++) для создания структуры стека из 10-ми элементов символьного типа.

#### 4.1.2. Код компетенции ОПК-3

##### 4.1.2.1 Типовые тестовые вопросы

№	Вопрос	Варианты ответов	
1	Отличительная особенность структуры В+-дерева состоит в том, что	1	эта структура позволяет получить максимально возможную степень ветвления во внутренних узлах
		+2	вся сопутствующая информация хранится в листьях, а во внутренних узлах хранятся только ключи и указатели на дочерние узлы

		3	каждый узел может иметь 2 или 3 потомка
2	При использовании методов хеширования задача поиска требуемой записи может быть сформулирована как	1	задача нахождения записи в таблице по ее номеру
		2	задача поиска записи в таблице с помощью специальной функции, которая называется хеш-функцией
		+3	задача нахождения в таблице по аргументу поиска $k$ адреса записи $a(k)$ , содержащей заданный ключ.
3	с помощью хеш-функции осуществляется	1	отыскание требуемой записи в таблице по заданному ключу записи.
		+2	отображение множества ключей всех записей на множество целых неотрицательных чисел.
		3	вычисление адреса записи в таблице по ее ключу.
4	Коллизией называется	1	ситуация, при которой двум одинаковым ключам могут соответствовать разные значения хеш-функции.
		2	ситуация, когда при выполнении хеш-адресации в одно и то же место таблицы должны быть помещены записи с разными ключами.
		+3	ситуация, при которой двум или более ключам соответствует одно и то же значение хеш-функции.
5	К хеш-функциям предъявляются следующие требования:	1	отсутствие коллизий и простота функций
		+2	простота функций, минимальное число коллизий и отсутствие явления скачивания записей в отдельных частях таблицы
		3	высокая скорость вычисления значений и отсутствие коллизий.
6	В методе деления по модулю значение хеш-функции вычисляется	1	ключ искомой записи делится на простое число, результат преобразуется к целому типу и используется как значение хеш-функции.

	следующим образом:	+2	ключ преобразуется в целое число, полученное значение делится на размер области значений адресов таблицы и в качестве значения хеш-функции используется остаток от деления.
		3	ключ преобразуется в целое число, делится на простое число и результат используется как значение хеш-функции.
7	При устранении коллизий методом цепочек выполняются следующие операции:	1	для каждого адреса хеш-таблицы создаются связные списки, в которых сохраняются записи с разными ключами.
		2	для записей с разными ключами образуются списки, где сохраняются значения записей.
		+3	в момент создания хеш-таблицы во всех ее записях размещаются указатели, которые используются для создания связных списков. В элементах списков сохраняются записи с разными ключами, для которых значения хеш-функции одинаковы.
8	Основная идея алгоритма обхода графа методом поиска в глубину состоит в том, что	1	выбирается первая вершина и помечается как посещенная. Затем последовательно просматриваются все остальные вершины графа.
		+2	посещается первая вершина, затем необходимо идти вдоль ребер графа, до попадания в тупик. После попадания в тупик нужно возвращаться назад вдоль пройденного пути, пока не будет обнаружена вершина, у которой есть еще не посещенная вершина, а затем необходимо двигаться в этом новом направлении.
		3	когда возможные пути по ребрам, выходящим из вершин, разветвляются, нужно сначала полностью исследовать одну ветвь и только потом переходить к другим ветвям (если они останутся нерассмотренными).
9	Основная идея алгоритма обхода графа методом поиска в ширину состоит в том, что	1	после посещения первой вершины, посещаются все соседние с ней вершины. Потом посещаются все вершины, находящиеся на расстоянии двух ребер от начальной. При каждом новом шаге посещаются вершины, расстояние от которых до начальной на единицу больше предыдущего.
		+2	сначала исследуется все вершины, смежные с начальной вершиной. Эти вершины находятся на расстоянии 1 от начальной. Затем исследуется все вершины на расстоянии 2 от начальной, затем все на расстоянии 3 и т.д.
		3	последовательно просматриваются все вершины графа, начиная с любой вершины, которая устанавливается как первая.
10	Алгоритм нахождения кратчайшего пути	1	алгоритм находит кратчайшее расстояние от одной из вершин графа до всех остальных

	Дейкстры отличается от алгоритма Флойда тем, что	2	он служит для нахождения кратчайших путей между всеми парами вершин графа
		+3	алгоритм находит кратчайшее расстояние от одной из вершин графа до всех остальных и работает только для графов без ребер отрицательного веса.

#### 4.1.3.2. Типовые теоретические вопросы

1. Сформулируйте понятие АВЛ-дерева поиска.
2. Поясните операцию малого правого (левого) вращения дерева.
3. Поясните операцию большого правого (левого) вращения дерева.
4. Поясните операцию добавления нового элемента в сбалансированное по АВЛ дерево.
5. Поясните операцию удаления элемента из сбалансированного по АВЛ дерева.
6. Сформулируйте понятие графа.
7. Поясните способы представления графов в памяти ЭВМ.
8. Поясните алгоритм поиска в глубину.
9. Сформулируйте понятия хеширования данных, хеш-функции и хеш-таблицы.
10. Поясните основные требования к хеш-функциям.
11. Какими свойствами должны быть наделены хеш-таблицы?
12. Поясните основные способы получения хеш-функций.
13. Что такое коллизии?
14. Сформулируйте известные способы устранения влияния коллизий.

#### 4.1.3.3 Типовые практические задачи

1. Составить БСА и подготовить программу (на языке Object Pascal или C++), в которой выполняется алгоритм обхода графа на основе поиска в глубину.
2. Составить БСА и реализовать программу (на языке Object Pascal или C++), в которой выполняется алгоритм обхода графа на основе поиска в ширину.
3. Используя обход графа в ширину, разработать алгоритм и подготовить программу (на языке Object Pascal или C++), для определения всех вершин графа, находящихся на фиксированном расстоянии  $d$  от данной вершины.
4. Перенумеровать вершины графа в порядке обхода в глубину и вычислить среднюю плотность графа как частное от деления количества его ребер на число вершин. Разработать алгоритм и подготовить программу(на языке Object Pascal или C++).

#### 4.1.3. Текущий контроль по дисциплине

##### Список вопросов к экзамену (зачету)

13. Понятие и общее описание структур данных (СД). Уровни представления СД.
14. Классификация СД. Операции над СД.
15. Статические СД. Векторы.
16. Массивы.
17. Записи и таблицы.
18. Линейные СД. Понятие списка.
19. Линейно-динамические СД. Одно- и двухсвязные списки.
20. Основные операции на связных списках.
21. Полустатические СД. Стеки и их применение в вычислительных системах.
22. Очереди и их применение в вычислительных системах.
23. Нелинейные связные СД. Сетевые и древовидные СД. Основные понятия.
24. Бинарные деревья. Представление бинарных деревьев в ОП.

25. Понятие и классификация алгоритмов обработки данных. Трудоемкость алгоритмов и методы ее оценки.
26. Методы оценки ресурсной эффективности алгоритмов. Базовые алгоритмы обработки данных.
27. Алгоритмы сортировки. Постановка задачи сортировки.
28. Основные алгоритмы внутренней сортировки.
29. Внешняя сортировка. Алгоритмы внешней сортировки слиянием.
30. Алгоритмы поиска данных. Постановка задачи поиска. Классификация алгоритмов поиска. Последовательный поиск.
31. Поиск в упорядоченном множестве записей. Бинарный поиск и его модификации.
32. Поиск по бинарному дереву.
33. Сбалансированные и оптимальные деревья поиска. АВЛ-деревья.
34. Сильно ветвящиеся деревья. Внешний поиск.
35. Понятия хэш-функции и хэш-адресации.
36. Методы построения хэш-функций.
37. Коллизии и способы их устранения.
38. Комбинированные способы построения таблиц на основе принципа хэш-адресации.  
Построение таблиц компиляторов.
39. Алгоритмы на графах. Способы представления графов в ЭВМ.
40. Обход вершин графов. Поиск в глубину.
41. Обход вершин графов. Поиск в ширину.
42. Задача нахождения пути в графах.
43. Построение стягивающего дерева графа.
44. Отыскание фундаментального множества циклов в графах.

Список вопросов к практическим занятиям

1. Сформулируйте понятие линейного списка.
2. Приведите логическую структуру односвязного линейного списка.
3. Поясните содержание основных операций односвязными списками.
4. В чем состоит особенность структуры двусвязных списков.
5. Поясните алгоритмы выполнения операций с двусвязными списками.
6. В чем состоит особенность структуры кольцевых списков.
7. Поясните алгоритмы выполнения операций с кольцевыми списками.
8. Поясните логическую структуру и принцип работы стека.
9. Приведите содержание основных операций со стеком.
10. Поясните логическую структуру и принцип работы очереди.
11. Опишите основные операции с очередями.
12. Поясните логическую структуру и принцип работы дека.
13. Приведите содержание основных операций с деком.
14. Поясните логическую структуру и основные элементы древовидных структур.
15. Что такое степень вершин дерева и степень дерева.
16. Поясните основные способы обхода деревьев.
17. Поясните логическую структуру бинарных деревьев.
18. Чем отличается сбалансированное бинарное дерево от почти сбалансированного.
19. В чем состоят основные операции, осуществляемые с бинарными деревьями.
20. Объясните основные свойства красно-черных деревьев.
21. Поясните основные типы и понятия внешней сортировки данных.
22. Объясните на примере алгоритм внешней сортировки простым слиянием.
23. В чем заключаются особенности алгоритма внешней сортировки естественным слиянием.
24. Сформулируйте в общем виде алгоритм поиска данных.

25. Поясните алгоритм последовательного линейного поиска.
26. В чем особенности поиска с барьером.
27. Поясните алгоритм бинарного поиска.
28. Что такое двоичное упорядоченное дерево?
29. Сформулируйте основные операции с упорядоченными деревьями.
30. Что такое случайные деревья поиска?
31. Поясните понятие идеально сбалансированного дерева.
32. Сформулируйте понятие АВЛ-дерева поиска.
33. Поясните операцию малого правого (левого) вращения дерева.
34. Поясните операцию большого правого (левого) вращения дерева.
35. Поясните операцию добавления нового элемента в сбалансированное по АВЛ дерево.
36. Поясните операцию удаления элемента из сбалансированного по АВЛ дерева.
37. Сформулируйте понятие графа.
38. Поясните способы представления графов в памяти ЭВМ.
39. Поясните алгоритм поиска в глубину.
40. Поясните алгоритм поиска в ширину.
41. Сформулируйте алгоритм Дейкстры поиска кратчайшего пути в графе.
42. Сформулируйте алгоритм Флойда поиска кратчайшего пути в графе.
43. Сформулируйте переборный алгоритм поиска кратчайшего пути в графе.
44. Сформулируйте понятия хеширования данных, хеш-функции и хеш-таблицы.
45. Поясните основные требования к хеш-функциям.
46. Какими свойствами должны быть наделены хеш-таблицы?
47. Поясните основные способы получения хеш-функций.
48. Что такое коллизии?
49. Сформулируйте известные способы устранения влияния коллизий.

## **5. Учебно-методическое обеспечение дисциплины**

### **5.1. Рекомендуемая литература**

#### **а) основная**

1. Ахо А., Хопкрофт Дж., Ульман Дж. Построение и анализ вычислительных алгоритмов. М.: Мир, 1979.
2. Ахо А., Хопкрофт Дж., Ульман Дж. Структуры данных и алгоритмы. М.: Вильяме, 2001. 384 с.
3. Бентли Д. Жемчужины творчества программистов. М.: Радио и связь, 1990.
4. Вирт Н. Алгоритмы + структуры данных = программы. М.: Мир, 1985.
5. Вирт Н. Алгоритмы и структуры данных. М: Мир, 1989. 360 с.
6. Грин Д., Кнут Д. Математические методы анализа алгоритмов. М: Мир, 1987.
7. Гудман С, Хидетниеми С. Введение в разработку и анализ алгоритмов. М.: Мир, 1981.
8. Дейкстра Э. Дисциплина программирования. М: Мир, 1978.
9. Кнут Д. Е. Искусство программирования: В 3 т. М.: Вильяме, 2000.
10. Кормен Т., Лейзерсон Ч., Ривест Р. Алгоритмы: Построение и анализ. М.: МЦНМО, 2001.
11. Лэгсам И., Огенстайн М. Структуры данных для персональных ЭВМ. М.: Мир, 1989. 586 с.
12. Юре О. Графы и их применение. М.: Мир, 1965.
13. Рейнгольд Э., Нивергельт Ю., Део Н. Комбинаторные алгоритмы. Теория и практика. М.: Мир, 1980.
14. Сибуя М, Ямamoto Т. Алгоритмы обработки данных. М.: Мир, 1986. 218 с.
15. Успенский В. А., Семенов А. Л. Теория алгоритмов: открытия и приложения. М.: Наука, 1987.

16. Харари Ф. Теория графов. М.: Мир, 1973.

**б) дополнительная**

1. Структуры и алгоритмы обработки данных/ В. А. Матьяш, В. А. Путилов, В. В. Фильчаков, С. В. Щекин. Апатиты: КФ ПетрГУ, 2000. 80 с.
2. Уайс М. А. Организация структур данных и решение задач на C++. /Эком, 2009.
3. Засорин С.В. Структуры и алгоритмы обработки данных/ Методические указания к курсовой работе. Рязань, РГРТУ, 2012.
4. Засорин С.В. Структуры и алгоритмы обработки данных/ Методические указания к лабораторным работам. Рязань, РГРТУ, 2012

**3.2. Ресурсы информационно-телекоммуникационной сети «Интернет»**

1. <http://algo.4u.ru/>
2. <http://algolist.manual.ru/>
3. <http://alplib.chat.ru/>
4. <http://algo.do.ru/>
5. <http://hcinsu.chat.ru/>
6. <http://algolist.da.ru/>
7. <http://progstone.narod.ru/links/wantalgo.html>
8. <http://www.sevmashvtuz.edu/links/algorithms.html>
9. <http://www.intuit.ru/department/algorithms/staldata/>

Настоящий документ составлен в соответствии с ФГОС +3 по направлению 02.03.03 «Математическое обеспечение и администрирование информационных систем».

Составил  
к.т.н., доцент Засорин С. В.

Документ рассмотрен и утвержден  
на заседании кафедры ЭВМ №\_\_\_\_«\_» 2023г. (протокол № )

Зав. Кафедрой ЭВМ  
проф. Костров Б.В.\_\_\_\_\_