ПРИЛОЖЕНИЕ

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«Рязанский государственный радиотехнический университет им. В.Ф. Уткина»

Кафедра «Систем автоматизированного проектирования
вычислительных средств»

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

по дисциплине

«Структуры и алгоритмы обработки данных»

Направление подготовки
09.03.01 – «Информатика и вычислительная техника»

Направленность (профиль) подготовки:
«Интеллектуальные системы и технологии»

Квалификация выпускника – бакалавр

Формы обучения – очная

Рязань 2026

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.

Контроль знаний обучающихся проводится в форме промежуточной аттестации.

Промежуточная аттестация проводится в форме зачета и экзамена. Форма проведения зачета – тестирование, письменный опрос по теоретическим вопросам и выполнение практических заданий. 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 выраженностью компетенций и является важным качественным ориентиром для самосовершенствования.

Уровень освоения компетенций, формируемых дисциплиной:

Описание критериев и шкалы оценивания тестирования:

Шкала оценивания	Критерий
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:

Шкала оценивания	Критерий
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

Описание критериев и шкалы оценивания практического задания:

Шкала оценивания	Критерий
3 балла (эталонный уровень)	Задача решена верно
2 балла (продвинутый уровень)	Задача решена верно, но имеются неточности в логике решения
1 балл (пороговый уровень)	Задача решена верно, с дополнительными наводящими вопросами преподавателя
0 баллов	Задача не решена

На промежуточную аттестацию в форме зачета выносится тест, один теоретический вопрос и одна задача. Максимально студент может набрать 9 баллов. Итоговый суммарный балл студента, полученный при прохождении промежуточной аттестации, переводится в традиционную форму по системе «зачтено» и «не зачтено».

Оценка «зачтено» выставляется студенту, который набрал в сумме от 3 до 9 баллов. Обязательным условием является выполнение всех предусмотренных в течение семестра лабораторных работ и практических заданий.

Оценка «не зачтено» выставляется студенту, который набрал в сумме менее 3 баллов или не выполнил всех предусмотренных в течение семестра лабораторных работ и практических заданий.

На промежуточную аттестацию в форме экзамена выносится тест и два теоретических вопроса. 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 6 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 3 до 5 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 3 баллов или не выполнил всех предусмотренных в течение семестра практических заданий.

3 ПАСПОРТ ОЦЕНОЧНЫХ МАТЕРИАЛОВ ПО ДИСЦИПЛИНЕ

Контролируемые разделы (темы) дисциплины	Код контролируемой компетенции (или её части)	Вид, метод, форма оценочного мероприятия
Тема 1. Введение в алгоритмы и структуры данных	ОПК-8, ПК-7	зачет

Тема 2. Простые базовые и статические структуры данных	ОПК-8, ПК-7	зачет
Тема 3. Полустатические структуры данных	ОПК-8, ПК-7	зачет
Тема 4. Динамические структуры данных	ОПК-8, ПК-7	зачет
Тема 5. Анализ сложности и эффективности алгоритмов обработки данных. Итерация и рекурсия.	ОПК-8, ПК-7	зачет
Тема 6. Алгоритмы сортировки и поиска данных	ОПК-8, ПК-7	зачет
Тема 7. Файловые структуры данных и алгоритмы их обработки	ОПК-8, ПК-7	зачет
Тема 8. Методы разработки алгоритмов	ОПК-8, ПК-7	экзамен
Тема 9. Алгоритмы хеширования и поиска данных в таблицах	ОПК-8, ПК-7	экзамен
Тема 10. Использование деревьев в прикладных алгоритмах	ОПК-8, ПК-7	экзамен
Тема 11. Алгоритмы на графах	ОПК-8, ПК-7	экзамен
Тема 12. Эвристические алгоритмы	ОПК-8, ПК-7	экзамен
Тема 13. Метод ветвей и границ в прикладных алгоритмах	ОПК-8, ПК-7	экзамен
Тема 14. Параллельные алгоритмы	ОПК-8, ПК-7	экзамен

4 ТИПОВЫЕ КОНТРОЛЬНЫЕ ЗАДАНИЯ ИЛИ ИНЫЕ МАТЕРИАЛЫ

4.1. Промежуточная аттестация в форме зачета

Код компетенции	**Результаты освоения ОПОП Содержание компетенций**
ОПК-8	Способен разрабатывать алгоритмы и программы, пригодные для практического применения
ОПК-8.1. Понимает требования к алгоритмам, суть процесса алгоритмизации задач	
ОПК-8.2. Выполняет разработку алгоритмического и программного обеспечения для решения прикладных задач	
ПК-7	Способен применять различные модели и (или) технологии обработки больших данных (BD-4, уровень Б)
ПК-7.1. Осуществляет выбор технологий обработки больших данных, приемлемых для создания прикладной системы ИИ с заданными требованиями (BD-4.1, уровень Б)	
ПК-7.2. Разрабатывает и отлаживает прикладные решения с элементами ИИ с применением различных технологий обработки данных (BD-4.2, уровень Б)	

Типовые тестовые вопросы

Наименование секции: Структуры данных

Вопрос 1. Как называется линейный список, в котором доступен только последний элемент?
Ответы:

1. Массив
+2. Стек
3. Очередь

Вопрос 2. В какой структуре данных доступ к элементам реализован по принципу FIFO? Ответы:
1. Список
2. Стек
+3. Очередь

Вопрос 3. В какой структуре данных доступ к элементам реализован по принципу LIFO? Ответы:
1. Список
+2. Стек
3. Очередь

Вопрос 4. Какая проблема исключается при реализации списка с помощью указателей? Ответы:
+1. Ограниченная длина
2. Медленная вставка элементов
3. Сложность поиска элемента

Вопрос 5. В какой программной реализации очереди необходимо перемещать оставшиеся элементы при удалении данных? Ответы:
1. На основе циклического массива
+2. На основе линейного массива
3. С использованием указателей

Вопрос 6. Что определяет понятие абстрактного типа данных? Ответы:
+1. Математическую модель задачи (или ее фрагмента) с набором операторов для работы
2. Набор возможных значений данных
3. Множество допустимых операций над данными

Вопрос 7. Выполнение какого оператора над однонаправленным списком, реализованным с помощью указателей в динамической памяти, зависит от его текущей длины? Ответы:
1. Добавление элемента в заданную позицию
+2. Определение позиции заданного элемента
3. Удаление элемента из заданной позиции

Вопрос 8. Для какой программной реализации очереди наиболее экономно используется оперативная память? Ответы:
1. На основе циклического массива
2. На основе линейного массива
+3. С использованием указателей

Вопрос 9. Как называется специальный тип списка, в котором вставка и удаление элементов производится только с одного конца списка? Ответы:
1. Циклический массив
2. Очередь
+3. Стек

Вопрос 10. Какие структуры данных не применяют для описания деревьев произвольного вида? Ответы:
+1. Узловое представление дерева
2. Списки потомков
3. Массив предков

Вопрос 11. Какое представление бинарного дерева требует наименьших затрат оперативной памяти? Ответы:
1. Узловое представление дерева
2. Списки потомков
+3. Массив предков

Вопрос 12. Какой вариант представления бинарного дерева в динамической памяти является наиболее распространенным? Ответы:
1. С указателями на предков
+2. С указателями на потомков
3. С указателями на предков и на потомков

Вопрос 13. Какой вариант представления дерева в динамической памяти позволяет описать дерево произвольного вида? Ответы:
+1. С указателями на предков
2. С указателями на потомков
3. С указателями на предков и на потомков
Вопрос 14. Какое матричное представление ориентированного графа является наиболее экономичным по затратам памяти? Ответы:
1. Матрица смежности
2. Матрица инцидентности
+3. Векторы смежности

Вопрос 15. Какое матричное представление графа является наиболее удобным с алгоритмической точки зрения? Ответы:
+1. Матрица смежности
2. Матрица инцидентности
3. Векторы смежности

Вопрос 16. Какие данные из перечисленных не сохраняются при каждом вызове рекурсивной подпрограммы? Ответы:
1. Параметры, передаваемые подпрограмме
2. Локальные переменные подпрограммы
+3. Переменные внешней программы

Вопрос 17. Какие структуры данных применяются для моделирования рекурсии в прикладных программах? Ответы:
1. Двунаправленные списки
+2. Стеки
3. Очереди

Вопрос 18. Пирамида является специальной формой помеченного бинарного дерева, которое используется в улучшенном алгоритме внутренней сортировки. Какое из перечисленных условий не предъявляется к пирамиде, если она имеет высоту h? Ответы:
1. Все конечные вершины (листья) должны иметь глубину h или h-1
2. Метка любой вершины должна быть больше или равна меток ее прямых потомков

+3. Каждая вершина, которая не является конечной (листом), должна иметь два прямых потомка

Вопрос 19. Какое матричное представление ориентированного графа является наиболее удобным для поиска начальных вершин, в которые не заходят дуги?
Ответы:
+1. Матрица смежности
2. Матрица инцидентности
3. Векторы смежности

Вопрос 20. Какие структуры данных включает узловое представление бинарного дерева? Ответы:
1. Связанные однонаправленные списки, реализованные в динамической памяти
+2. Несколько одномерных массивов
3. Стековые наборы данных

Типовые практические задания

Задание 1. Счет в банке представляет собой структуру данных с полями: номер счета, код счета, фамилия владельца, сумма на счете, дата открытия счета, годовой процент начисления. Реализовать на основе линейного списка поиск и сортировку по номеру счета, дате его открытия и фамилии владельца.

Задание 2. Запись о товаре на складе представляет собой структуру данных с полями: номер склада, код товара, наименование товара, дата поступления на склад, срок хранения в днях, количество единиц товара, цена за единицу товара. Реализовать на основе линейного списка поиск и сортировку по номеру склада, наименованию товара. Вывести список просроченных товаров (поиск всех товаров, у которых на текущую дату истек срок хранения).

Задание 3. Запись о преподаваемой дисциплине представляется структурой данных: код дисциплины в учебном плане, наименование дисциплины, фамилия преподавателя, код группы, количество студентов в группе, количество часов лекций, количество часов практических занятий, вид итогового контроля (зачет или экзамен), дата начала занятий. Реализовать на основе линейного списка поиск и сортировку по фамилии преподавателя, количеству часов, дате начала занятий.

Задание 4. Информационная запись о книге, выданной на руки абоненту, представляет собой структуру данных следующего вида: номер читательского билета, фамилия абонента, дата выдачи, количество дней, автор, название, год издания, цена. Реализовать на основе линейного списка поиск и сортировку по номеру читательского билета, автору книги. Вывести список всех просроченных книг (поиск всех книг, которые на текущую дату должны быть сданы).

Задание 5. Информационная запись о файле содержит следующие поля: каталог, имя файла, расширение, дата и время создания, атрибуты «только для чтения», «скрытый», «системный», количество выделенных секторов (размер сектора принять равным 512 байтам). Реализовать на основе линейного списка поиск и сортировку по каталогу, дате создания файла. Выяснить, поместится ли файл на носитель с некоторым количеством секторов.

Задание 6. Разовый платеж за телефонный разговор представлен структурой данных с полями: фамилия плательщика, номер телефона, дата разговора, тариф за минуту разговора, время начала разговора, время окончания разговора. Реализовать на основе линейного списка

поиск и сортировку по фамилии плательщика, дате разговора. Найти все разговоры со временем разговора больше заданного.

Задание 7. Модель компьютера характеризуется кодом и маркой компьютера, типом процессора (может содержать цифры и буквы), частотой работы процессора, объемом оперативной памяти, объемом жесткого диска, датой выпуска на рынок, стоимостью компьютера в рублях и количеством экземпляров, имеющихся в наличии. Реализовать на основе линейного списка поиск и сортировку по типу процессора, объему ОЗУ, дате выпуска компьютера на рынок.

Задание 8. Список абонентов сети кабельного телевидения состоит из элементов следующей структуры данных: фамилия, район, адрес, теле фон, номер договора, дата заключения договора, оплата установки, дата последнего платежа. Реализовать на основе линейного списка поиск и сортировку по району, номеру договора, дате последнего платежа.

Задание 9. Сотрудник некоторой организации представлен структурой данных с полями: табельный номер, номер отдела, фамилия, оклад, дата поступления на работу, процент надбавки, процент налоговых сборов, количество отработанных дней в месяце, количество рабочих дней в месяце, начислено, удержано. Реализовать на основе линейного списка поиск и сортировку по номеру отдела, дате поступления на работу, фамилии.

Задание 10. Запись о багаже пассажира авиарейса содержит следующие поля: номер рейса, дата и время вылета, пункт назначения, фамилия пассажира, количество мест багажа, суммарный вес багажа. Реализовать на основе линейного списка поиск и сортировку по дате вылета, пункту назначения. Найти всех пассажиров, у которых масса багажа выше максимально допустимой.

Задание 11. Учетная запись посещения спорткомплекса имеет структуру: фамилия клиента, код и вид спортивного занятия, фамилия тренера, дата и время начала тренировки, количество минут, тариф. Реализовать на основе линейного списка поиск и сортировку по фамилии клиента, дате начала и количеству минут тренировки (больше или меньше введенного).

Задание 12. Одна запись о медикаменте содержит следующие поля: номер аптеки, название лекарства, количество упаковок, имеющиеся в наличии в данной аптеке, стоимость одной упаковки, дата поступления в аптеку, срок хранения (в днях). Реализовать на основе линейного списка поиск и сортировку по номеру аптеки, наименованию препарата, дате поступления.

Задание 13. Одна запись журнала учета содержит поля: код игрушки, название игрушки, тип игрушки, возрастные ограничения (например, от 6 лет), цена за единицу, количество в наличии, да та поступления в магазин, поставщик. Реализовать на основе линейного списка поиск и сортировку по дате поступления, поставщику, возрастным ограничениям.

Задание 14. Один элемент (автомобиль) представляет собой структуру данных с полями: фамилия владельца, марка автомобиля, требуемая марка бензина, мощность двигателя, объем бака, остаток бензина, объем масла, дата техосмотра. Дана фиксированная цена литра бензина и заливки масла. Реализовать на основе линейного списка поиск и сортировку по марке автомобиля, мощности двигателя, дате техосмотра.

Задание 15. Запись в журнале зимней экзаменационной сессии представляет собой структуру данных с полями: курс, код группы, фамилия студента, дата поступления, номер зачетной книжки, дисциплина, оценка за экзамен по дисциплине. Реализовать на основе ли-

нейного списка поиск и сортировку по номеру курса, номеру зачетной книжки, дате поступления.

Задание 16. Структура одной записи оплаты за коммунальные услуги содержит поля: номер дома, номер квартиры, фамилия владельца, вид платежа (квартплата, газ, вода, электричество), дата платежа, сумма платежа, процент пени, на сколько дней просрочен, платеж. Реализовать на основе линейного списка поиск и сортировку по номеру дома, виду платежа, дате платежа.

Задание 17. Одна запись счета за ремонтные работы содержит поля: название фирмы, вид работ, единица измерения, стоимость единицы выполненных работ, дата исполнения, количество выполненных работ. Реализовать на основе линейного списка поиск и сортировку по названию фирмы, виду работ, дате исполнения.

Задание 18. Одна учетная запись журнала стоянки автомобилей имеет структуру: номер автомобиля, фамилия владельца, дата и время начала, дата и время окончания, тариф за час. Реализовать на основе линейного списка поиск и сортировку по номеру автомобиля, дате начала стоянки, фамилии владельца.

Задание 19. Структура записи о сельскохозяйственном продукте содержит поля: наименование района (где выращивают), наименование продукта, площадь (га), урожайность (кг/га), цена за 1 кг, потери при транспортировке (%), стоимость продукта, предполагаемая дата сбора. Реализовать на основе линейного списка поиск и сортировку по наименованию района, урожайности, предполагаемой дате сбора.

Задание 20. В туристической фирме учетная запись о проданном туре содержит следующие поля: наименование тура, фамилия клиента, цена одного дня (в рублях), дата заезда, количество дней, стоимость проезда, курс валюты, количество валюты, стоимость проезда. Реализовать на основе линейного списка поиск и сортировку по наименованию тура, стоимости проезда, дате заезда.

Задание 21. Сотовый телефон характеризуется названием производителя, номером модели (может содержать цифры и буквы), временем работы аккумулятора, наличием и максимальной емкостью карты памяти, датой выпуска на рынок, стоимостью в рублях и количеством экземпляров, имеющихся в наличии. Реализовать на основе линейного списка поиск и сортировку по номеру модели, объему памяти на карте, дате вы пуска на рынок.

Задание 22. Одна запись о предмете мебели содержит следующие поля: артикул (может содержать цифры и буквы), наименование, цвет, стоимость, дата изготовления, количество имеющихся в наличии экземпляров. Реализовать на основе линейного списка поиск и сортировку по артикулу, количеству экземпляров, дате изготовления.

Типовые теоретические вопросы

1. Парадигмы программирования. Методологии императивного, структурного и процедурного программирование.
2. Понятие алгоритма. Свойства и виды алгоритмов. Базовые канонические структуры алгоритмов.
3. Этапы построения и способы представления алгоритмов.
4. Понятие структуры данных. Классификация структур данных.
5. Концепция абстрактного типа данных.

6. Простые базовые структуры данных. Представление в памяти и операции над данными.
7. Статические структуры данных. Представление в памяти и операции над данными.
8. Полустатические структуры данных. Представление в памяти и операции над данными.
9. Динамические структуры данных. Общая характеристика и классификация.
10. Абстрактный тип данных «Список». Реализация с использованием массива.
11. Абстрактный тип данных «Список». Реализация с использованием указателей.
12. Абстрактный тип данных «Стек». Реализация с использованием массива.
13. Абстрактный тип данных «Очередь». Реализация в динамической памяти.
14. Абстрактный тип данных «Очередь». Реализация с помощью циклического массива.
15. Машинное представление деревьев с использованием массивов.
16. Машинное представление деревьев в динамической памяти.
17. Машинное представление графов с использованием матриц.
18. Машинное представление графов с использованием списков.
19. Задача сортировки. Характеристика используемых методов.
20. Прямые методы сортировки.
21. Быстрая сортировка.
22. Сортировка Шелла.
23. Пирамидальная сортировка.
24. Линейный и бинарный поиск.
25. Внешняя сортировка. Прямое слияние.
26. Внешняя сортировка. Естественное слияние.
27. Хранение и поиск данных в файлах. Индексирование файлов.

4.1. Промежуточная аттестация в форме экзамена

Код компетенции	Результаты освоения ОПОП Содержание компетенций
ОПК-8	Способен разрабатывать алгоритмы и программы, пригодные для практического применения
ОПК-8.1. Понимает требования к алгоритмам, суть процесса алгоритмизации задач	
ОПК-8.2. Выполняет разработку алгоритмического и программного обеспечения для решения прикладных задач	

Типовые тестовые вопросы

Наименование секции: Алгоритмы обработки данных

Вопрос 21. Как называется сортировка, которая выполняется только в оперативной памяти компьютера? Ответы:
1. Быстрая
+2. Внутренняя
3. Внешняя

Вопрос 22. В каком случае улучшенные методы сортировки имеют значительное преимущество? Ответы:
+1. При большом количестве сортируемых элементов
2. Когда сортируемые элементы обратно упорядочены
3. При малом количестве сортируемых элементов

Вопрос 23. Как называется алгоритм сортировки, для которого все перестановки элементов выполняются строго в одном массиве? Ответы:

1. Строгий
+2. Прямой
3. Улучшенный

Вопрос 24. В каком прямом методе сортировки осуществляется систематическое сравнение ключей соседних элементов? Ответы:
+1. Сортировка обменами
2. Сортировка выбором
3. Сортировка простыми вставками

Вопрос 25. Какой прямой метод сортировки является наиболее быстрым для почти упорядоченного массива сортируемых элементов? Ответы:
1. Сортировка обменами
2. Сортировка выбором
+3. Сортировка простыми вставками

Вопрос 26. Какой прямой метод сортировки лежит в основе алгоритма Шелла? Ответы:
1. Сортировка обменами
2. Сортировка выбором
+3. Сортировка простыми вставками

Вопрос 27. Какую оценку вычислительной сложности имеют прямые методы сортировки массивов? Ответы:
1. $O(n)$
+2. $O(n^2)$
3. $O(n\log_2 n)$

Вопрос 26. Какой алгоритм обхода вершин бинарного дерева начинает свою работу с просмотра корня? Ответы:
+1. Обход в прямом порядке
2. Обход в обратном порядке
3. Симметричный обход

Вопрос 28. Какой алгоритм обхода вершин бинарного дерева завершает свою работу просмотром корня? Ответы:
1. Обход в прямом порядке
+2. Обход в обратном порядке
3. Симметричный обход

Вопрос 29. Какой алгоритм обхода вершин бинарного дерева можно использовать для сортировки? Ответы:
1. Обход в прямом порядке
2. Обход в обратном порядке
+3. Симметричный обход
Вопрос 30. Что является целью топологической сортировки вершин ориентированного графа? Ответы:
1. Расположение вершин графа в линейном порядке
+2. Получение такой нумерации вершин, для которой любая дуга направлена от вершины с меньшим индексом к вершине с большим индексом
3. Получение такой нумерации вершин, для которой любая дуга направлена от вершины с большим индексом к вершине с меньшим индексом

Вопрос 31. Какой алгоритм поиска является наиболее эффективным для упорядоченного списка ключей? Ответы:
1. Линейный поиск
+2. Бинарный (логарифмический) поиск
3. Поиск по бинарному дереву

Вопрос 32. Какой алгоритм поиска является наиболее эффективным для неупорядоченного (случайного) списка ключей? Ответы:
1. Линейный поиск
2. Бинарный (логарифмический) поиск
+3. Поиск по бинарному дереву

Вопрос 33. В каком методе поиска может использоваться следующий фрагмент программы:

```
repeat
   i:=i+1
until (a[i]=x) or (i=n);
```

Ответы:
1. Логарифмический
+2. Линейный
3. По бинарному дереву

Вопрос 34. В каком методе поиска может использоваться следующий фрагмент программы:

```
repeat
   k:=(i+j)div2;
   if x>a[k] then i:=k+1 else j:=k-1;
until (a[k]=x) or (i>j);
```

Ответы:
+1. Бинарный
2. Линейный
3. Интерполяционный

Вопрос 35. Что означает устойчивость сортировки? Ответы:
+1. Сохранение относительного порядка записей с одинаковыми значениями ключей
2. Отсутствие записей с одинаковыми значениями ключей
3. Процесс сортировки уже упорядоченного набора записей

Вопрос 36. Какой метод обхода вершин ориентированного графа может использоваться для нахождения контуров в этом графе? Ответы:
1. Линейный поиск
+2. Поиск в глубину
3. Поиск в ширину

Вопрос 37. Сколько раз просматривается каждая вершина ориентированного графа при реализации алгоритма поиска в глубину? Ответы:
1. Заранее неизвестно, так как зависит от структуры графа
+2. Один
3. Равно числу компонент связности графа

Вопрос 38. Чему равно количество внешних вызовов процедуры поиска в глубину при обходе всех вершин неориентированного графа? Ответы:
1. Заранее неизвестно, так как зависит от структуры графа
2. Один вызов

+3. Равно числу компонент связности графа

Вопрос 39. Чему равно количество внешних вызовов процедуры поиска в глубину при обходе всех вершин ориентированного графа? Ответы:
+1. Заранее неизвестно, так как зависит от структуры графа
2. Один вызов
3. Равно числу компонент связности графа

Вопрос 40. Чему равно количество внешних вызовов процедуры поиска в ширину при обходе вершин неориентированного графа? Ответы:
1. Заранее неизвестно, так как зависит от структуры графа
2. Один вызов
+3. Равно числу компонент связности графа

Вопрос 41. Сколько раз просматривается каждая вершина ориентированного графа при реализации алгоритма поиска в глубину? Ответы:
1. Заранее неизвестно, так как зависит от структуры графа
+2. Один
3. Равно числу компонент связности графа

Вопрос 42. Какой алгоритм позволяет построить транзитивное замыкание матрицы смежности ориентированного графа? Ответы:
1. Алгоритм Прима
2. Алгоритм Дейкстры
+3. Алгоритм Уоршелла

Вопрос 43. Что является основой алгоритмов внешней сортировки? Ответы:
1. Процедура разделения сортируемого набора данных
+2. Процедура слияния пары упорядоченных наборов данных
3. Процедура сортировки набора данных прямым методом

Вопрос 44. Как оценивается вычислительная сложность алгоритма внешней сортировки прямым слиянием? Ответы:
1. $O(n)$
2. $O(n^2)$
+3. $O(n\log_2 n)$

Вопрос 45. Как оценивается вычислительная сложность внешней сортировки для алгоритма естественного слияния? Ответы:
1. $O(n)$
+2. $O(n\log_2 n)$
3. $O(n^2)$

Вопрос 46. В каком виде хранятся записи основного файла с данными при реализации поиска с помощью разреженного индексного файла? Ответы:
+1. В отсортированном по значениям ключей виде
2. В порядке их записи в файл
3. В случайном порядке

Вопрос 47. Какое назначение имеет индексный файл при внешнем поиске данных? Ответы:
1. Уменьшение объема памяти, занимаемой основным файлом с данными
+2. Ускорение поиска записей в основном файле данных

3. Хранение информации о характеристиках основного файла с данными

Вопрос 48. Какие дополнительные структуры данных используются для реализации внешнего поиска в файле с неотсортированными записями? Ответы:
+1. Плотный индексный файл
2. Разреженный индексный файл
3. Разреженный и плотный индексные файлы

Вопрос 49. Какие факторы влияют на эффективность внешней сортировки по методу естественного слияния? Ответы:
1. Быстродействие устройства хранения данных
+2. Частичная упорядоченность ключей сортируемых данных
3. Количество записей в сливаемых сериях

Вопрос 50. Какой метод обхода вершин ориентированного бесконтурного графа может применяться для решения задачи топологической сортировки? Ответы:
+1. Поиск в глубину
2. Поиск в ширину
3. Логарифмический поиск

Типовые теоретические вопросы

1. Рекурсия и итерация в алгоритмах обработки данных.
2. Анализ сложности алгоритмов. Временная и емкостная сложность.
3. Полиномиальные и экспоненциальные алгоритмы. NP-полные задачи.
4. Методы разработки алгоритмов. Эвристические алгоритмы. Жадные алгоритмы.
5. Алгоритмы обхода бинарных деревьев.
6. Сортировка с использованием обхода бинарного дерева.
7. Поиск по бинарному дереву.
8. Алгоритмы поиска в тексте.
9. Хеширование данных. Перемешанные таблицы.
10. Поиск в хеш-таблицах. Методы разрешения коллизий.
11. Процедура поиска в глубину на графе.
12. Нахождение связных компонент неориентированного графа.
13. Топологическая сортировка вершин ориентированного графа.
14. Процедура поиска в ширину на графе.
15. Поиск транзитивного замыкания для ориентированных графов.
16. Алгоритмы построения кратчайших путей на графах. Обзор и сравнительный анализ.
17. Построение кратчайшего пути на графе методом динамического программирования.
18. Алгоритм Дейкстры построение кратчайшего пути на графе.
19. Алгоритм Форда-Беллмана построение кратчайшего пути на графе.
20. Алгоритм Флойда построения кратчайших путей между всеми парами вершин.
21. Алгоритмы построения критического пути на графе.
22. Алгоритмы построения минимального остовного дерева.
23. Волновой алгоритм и его модификации.
24. Эвристические алгоритмы составления расписаний.
25. Эвристические алгоритмы решения задачи упаковки.
26. Алгоритмы сжатия данных.
27. Применение жадных алгоритмов для приближенного решения сложных задач.
28. Общая схема метода ветвей и границ и его приложения.
29. Алгоритм решения задачи коммивояжера методом ветвей и границ.

30. Параллельные архитектуры компьютеров и модели параллельных вычислений.
31. Алгоритмы параллельной сортировки и поиска данных.
32. Параллельные численные алгоритмы.
33. Параллельные алгоритмы на графах.

Типовые вопросы к лабораторным и практическим занятиям

1. Чем вызвана необходимость представления разреженных матриц в упакованном виде?
2. Какая матрица называется разреженной?
3. Как осуществляется упаковки разреженной матрицы в форму таблицы связей?
4. В чем заключаются преимущества и недостатки представления матрицы в виде таблицы связей?
5. Какой принцип используется при упаковке матрицы по строчной схеме?
6. Как осуществляется сжатие матрицы по столбцовой схеме упаковки?
7. Что такое структура смежности?
8. В чем заключаются преимущества и недостатки структур смежности, используемых при упаковке матриц?
9. Для чего предназначены списки указателей, используемые в структурах смежности?
10. В чем заключается строчно-столбцовый метод упаковки матриц и каковы его преимущества и недостатки?
11. Какой метод упаковки позволяет достичь максимальной степени сжатия информации исходной матрицы и каковы принципы его работы?
12. Что показывают элементы массивов в модифицированной строчно-столбцовой схеме упаковки?
13. Как формулируется задача сортировки и какова цель ее решения?
14. Что такое функция сортировки?
15. В чем заключается устойчивость сортировки?
16. В чем состоит принципиальное отличие алгоритмов внутренней и внешней сортировки?
17. Как определяется понятие «минимальная по памяти сортировка»?
18. Почему алгоритм сортировки обменами также называется «пузырьковой сортировкой»?
19. Как можно ускорить работу алгоритма сортировки обменами?
20. В чем заключается сущность алгоритма сортировки простыми вставками?
21. Как можно ускорить работу алгоритма сортировки простыми вставками?
22. Как изменится алгоритм сортировки выбором при необходимости упорядочения элементов массива по убыванию?
23. Сколько раз просматривается упорядочиваемый массив в алгоритме сортировки выбором?
24. По каким параметрам оценивается эффективность алгоритмов сортировки?
25. Как формулируется задача поиска данных и каковы возможные результаты ее решения?
26. Каким образом связаны задачи поиска и сортировки данных?
27. Что такое аргумент поиска?
28. Каковы возможные варианты реализации алгоритма линейного поиска?
29. Как оценивается эффективность алгоритма линейного поиска?
30. В чем отличие итерационного и рекурсивного вариантов алгоритма бинарного поиска?
31. Как оценивается эффективность алгоритма бинарного поиска?
32. На каких основных положениях основан алгоритм поиска по бинарному дереву?
33. Как строится бинарное дерево для организации поиска данных?

34. В чем состоят преимущества алгоритма поиска по дереву в сравнении с другими алгоритмами поиска?
35. Как оценивается эффективность алгоритма поиска по бинарному дереву?
36. При каких условиях обеспечивается эффективное использование алгоритмов поиска данных?
37. На каких положениях основан алгоритм поиска в глубину на ориентированных графах?
38. Каким образом можно использовать алгоритм поиска в глубину для решения задачи топологической сортировки вершин ориентированного графа?
39. Как определяется DFS-дерево, формируемое алгоритмом поиска в глубину на графе?
40. Какие свойства DFS-дерева используются при разработке прикладных алгоритмов на основе процедуры поиска в глубину?

Типовые задания для практической и самостоятельной работы

Практические задания

1. Программная реализация и исследование алгоритмов внешней сортировки данных.
2. Программная реализация и исследование вычислительной сложности алгоритмов обработки данных.
3. Разработка и реализация в динамической памяти абстрактных типов данных «Очередь» и «Двунаправленный список».
4. Программная реализация и исследование алгоритма поиска в ширину на неориентированных графах.
5. Программная реализация и исследование алгоритма поиска в ширину на ориентированных графах.
6. Разработка программы определения наличия контуров в ориентированных графах.
7. Разработка программы определения наличия контуров в неориентированных графах.
8. Программная реализация и исследование алгоритма быстрой сортировки.
9. Программная реализация и исследование алгоритма сортировки Шелла.
10. Программная реализация и исследование алгоритма пирамидальной сортировки.

Теоретические задания (*темы рефератов*)

1. Методы оценки вычислительной сложности алгоритмов обработки данных.
2. Экстремальные числа графов и их применение в алгоритмах решения прикладных задач.
3. NP-полные задачи обработки данных.
4. Сравнительный анализ алгоритмов обработки данных с полиномиальной и экспоненциальными оценками вычислительной сложности.
5. Управление памятью в прикладных алгоритмах.
6. Сбалансированные деревья.
7. Алгоритмы внешнего поиска данных.
8. Алгоритмы внешнего поиска данных.
9. Рекурсия и итерация в алгоритмах обработки данных.
10. Абстрактные типы данных и их применении в прикладных программах.

Составил
проф. кафедры САПР ВС,
д.т.н., проф. С.В. Скворцов

Оператор ЭДО ООО "Компания "Тензор"
ДОКУМЕНТ ПОДПИСАН ЭЛЕКТРОННОЙ ПОДПИСЬЮ
ПОДПИСАНО ЗАВЕДУЮЩИМ КАФЕДРЫ — ФГБОУ ВО "РГРТУ", РГРТУ, Корячко Вячеслав Петрович, Заведующий кафедрой САПР — 09.10.26 14:58 (MSK) — Простая подпись
ПОДПИСАНО ЗАВЕДУЮЩИМ ВЫПУСКАЮЩЕЙ КАФЕДРЫ — ФГБОУ ВО "РГРТУ", РГРТУ, Корячко Вячеслав Петрович, Заведующий кафедрой САПР — 09.10.26 14:58 (MSK) — Простая подпись