

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«РЯЗАНСКИЙ ГОСУДАРСТВЕННЫЙ РАДИОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
ИМЕНИ В.Ф. УТКИНА»

Кафедра «Вычислительной и прикладной математики»

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ДИСЦИПЛИНЫ

«Алгоритмы и структуры данных»

Специальность

09.05.01 «Применение и эксплуатация систем специального назначения»

Направленность (профиль) подготовки

«Математическое_ программное и информационное обеспечение
вычислительной техники и автоматизированных систем»

Уровень подготовки
специалитет

Квалификация выпускника – инженер

Форма обучения – очная

Рязань

ОБЩИЕ ПОЛОЖЕНИЯ

Оценочные материалы – это совокупность учебно-методических материалов и процедур, предназначенных для оценки качества освоения обучающимися данной дисциплины как части основной образовательной программы.

Цель – оценить соответствие знаний, умений и уровня приобретенных компетенций, обучающихся целям и требованиям основной образовательной программы в ходе проведения текущего контроля и промежуточной аттестации.

Основная задача – обеспечить оценку уровня сформированности общекультурных и профессиональных компетенций, приобретаемых обучающимся в соответствии с этими требованиями.

Контроль знаний, обучающихся проводится в форме текущего контроля и промежуточной аттестации.

Текущий контроль успеваемости и промежуточная аттестация проводятся с целью определения степени усвоения учебного материала, своевременного выявления и устранения недостатков в подготовке обучающихся, организации работы обучающихся в ходе учебных занятий и оказания им индивидуальной помощи.

К контролю текущей успеваемости относятся проверка знаний, умений и навыков, обучающихся на лабораторных работах по результатам выполнения и защиты обучающимися индивидуальных заданий, по результатам выполнения контрольных работ и тестов, по результатам проверки качества конспектов лекций и иных материалов.

В качестве оценочных средств на протяжении семестра используются устные и письменные ответы студентов на индивидуальные вопросы, письменное тестирование по теоретическим разделам курса, реферат. Дополнительным средством оценки знаний и умений студентов является отчет о проведении лабораторных работ и их защита.

По итогам курса обучающиеся сдают зачет. Форма проведения – устный ответ с письменным подкреплением по утвержденным билетам, сформулированным с учетом содержания дисциплины. В билет для зачета включается два теоретических вопроса. В процессе подготовки к устному ответу студент должен составить в письменном виде план ответа.

2. ПАСПОРТ ОЦЕНОЧНЫХ МАТЕРИАЛОВ ПО ДИСЦИПЛИНЕ(МОДУЛЮ)

№ п/п	Контролируемые разделы дисциплины	Код контролируемой компетенции (или её части)	Наименование оценочного средства
1	Введение в дисциплину	ОПК-3, ПК-3, ПК-4	Зачет
2	Линейные структуры данных	ОПК-3, ПК-3, ПК-4	Зачет
2.1	Стеки, очереди, деки	ОПК-3, ПК-3, ПК-4	Зачет

2.2	Линейные списки	ОПК-3, ПК-3, ПК-4	Зачет
3	Нелинейные структуры данных	ОПК-3, ПК-3, ПК-4	Зачет
3.1	Бинарные деревья	ОПК-3, ПК-3, ПК-4	Зачет
3.2	Деревья поиска	ОПК-3, ПК-3, ПК-4	Зачет
3.3	Графы	ОПК-3, ПК-3, ПК-4	Зачет
3.4	Алгоритмы на графах	ОПК-3, ПК-3, ПК-4	Зачет
4	Сортировка	ОПК-3, ПК-3, ПК-4	Зачет
4.1	Внутренняя сортировка	ОПК-3, ПК-3, ПК-4	Зачет
4.2	Внешняя сортировка	ОПК-3, ПК-3, ПК-4	Зачет
5	Поиск и кодирование данных	ОПК-3, ПК-3, ПК-4	Зачет
5.1	Поиск	ОПК-3, ПК-3, ПК-4	Зачет
5.2	Расстановка	ОПК-3, ПК-3, ПК-4	Зачет
6	Теория сложности алгоритмов	ОПК-3, ПК-3, ПК-4	Зачет

2. ОПИСАНИЕ ПОКАЗАТЕЛЕЙ И КРИТЕРИЕВ ОЦЕНИВАНИЯ КОМПЕТЕНЦИЙ

Сформировать каждой компетенции в рамках освоения данной дисциплины оценивается по трехуровневой шкале:

- пороговый уровень (удовлетворительный) является обязательным для всех обучающихся по завершении освоения дисциплины;
- продвинутый уровень (хороший) характеризуется превышением минимальных характеристик сформированности компетенций по завершении освоения дисциплины;
- эталонный уровень (отличный) характеризуется максимально возможной выраженностью компетенций и является важным качественным ориентиром для самосовершенствования обучаемого.

При достаточном качестве освоения более 81% приведенных знаний, умений и навыков преподаватель оценивает освоение данной компетенции в рамках настоящей дисциплины на эталонном уровне, при освоении более 61% приведенных знаний, умений и навыков — на продвинутом, при освоении более 41% приведенных знаний, умений и навыков — на пороговом уровне. При освоении менее 40% приведенных знаний, умений и навыков компетенция в рамках настоящей дисциплины считается неосвоенной.

Уровень сформированности каждой компетенции на различных этапах ее формирования в процессе освоения дисциплины оценивается в ходе текущего контроля успеваемости и представлено различными видами оценочных средств.

Учитываются:

- уровень усвоения материала, предусмотренного программой курса;
- умение анализировать материал и устанавливать причинно-следственные связи;
- ответы на вопросы: полнота, аргументированность, убежденность, качество ответа (его общая композиция, логичность, общая эрудиция);

- использование основной и дополнительной литературы при подготовке, и принимаются во внимание знания, умения, навыки, перечисленные в п. 2 рабочей программы дисциплины.

Критерии оценивания уровня сформированности компетенции в процессе выполнения контрольных заданий:

41%-60% правильных ответов соответствует пороговому уровню сформированности компетенции на данном этапе ее формирования;

61%-80% правильных ответов соответствует продвинутому уровню сформированности компетенции на данном этапе ее формирования;

81%-100% правильных ответов соответствует эталонному уровню сформированности компетенции на данном этапе ее формирования.

Сформированность уровня компетенций не ниже порогового является основанием для допуска обучающегося к промежуточной аттестации по данной дисциплине.

Формой промежуточной аттестации по данной дисциплине является зачет, оцениваемый по принятой в ФГБОУ ВО «РГРТУ» двухбалльной системе:

Шкала оценивания	Критерии оценивания
«зачтено»	студент должен: продемонстрировать всесторонние, систематические и глубокие знания учебного материала. Проверочную графическую работу на зачете – выполнил уверенно и без ошибок. (могут быть незначительные ошибки и исправленные самим студентом без помощи преподавателя). Студент осмысленно и достаточно глубоко освоил стандарты ЕСКД, уверенно и без ошибок отвечает на вопросы. Все графические построения, сделанные в семестре, – правильные и четкие. Графическое оформление и надписи выполнены без нарушения ГОСТ. Владеет навыками и приемами черчения.
«не зачтено»	«не зачтено»: ставиться за один из указанных недостатков. Зачетная работа выполнена с ошибками, после наводящих вопросов преподавателя студент не исправляет ошибки в зачетной работе. Слабые знания теории, основных положений ГОСТ и неумение применять их на практике. В работах, выполненных в семестре, были серьезные недоработки в оформлении чертежей (толщина обводки и структура многих линий, надписи выполнены со значительным отступлением от ГОСТ. Допущены грубые ошибки, связанные с выполнением задания (количество видов, сечений и разрезов недостаточно, формы отдельных деталей нельзя установить по чертежу и т.п.).

4. ТИПОВЫЕ КОНТРОЛЬНЫЕ ЗАДАНИЯ ИЛИ ИНЫЕ МАТЕРИАЛЫ

4.1. Контрольные вопросы и типовые задания текущего контроля для практических занятий и при защите лабораторных работ

Стеки, очереди, деки

1. В чём заключаются достоинства и недостатки последовательного и связанного способов реализации динамических структур данных? Назовите принципы

функционирования стека, очереди и дека.

2. Приведите примеры использования стека в программировании.

3. Реализуйте класс – стек с базовым набором методов на основе массива нетипированных указателей на размещенные в динамической памяти элементы.

4. С использованием основных методов работы со стеком составить программу копирования элементов стека в новый стек в том же порядке.

5. Преобразуйте инфиксное выражение $(ax^2 + 1)/(bx + a) \div abx$ в префиксную форму записи

при условии, что все идентификаторы – однобуквенные.

6. Преобразуйте инфиксное выражение $(a * (b \div c) \div d) / 2$ в постфиксную форму записи

при условии, что все идентификаторы – однобуквенные.

7. Опишите процесс функционирования стека при вычислении значения постфиксного

выражения $abc \div d \div 2 /$ при $a \div 2, b \div 5, c \div 3, d \div 4$.

8. Реализуйте метод копирования элементов очереди в новую очередь.

9. С использованием стандартного набора методов составьте программу копирования элементов стека в новый стек в том же порядке.

Линейные списки

1. Реализуйте метод переноса элементов односвязного списка с заданным значением в новый список того же типа.

2. Реализуйте метод удаления предшествующего элемента односвязного списка для элемента с заданным значением.

3. Реализуйте метод удаления двух соседних элементов двусвязного списка для элемента с заданным значением.

4. Реализуйте метод возвращения значения k -го от начала элемента двусвязного списка.

5. Реализуйте метод возвращения адреса k -го от конца элемента двусвязного списка.

6. С использованием базового набора методов составьте программу копирования из односвязного списка строк в двусвязный список строк, начинающихся на буквы «Б» или «б».

7. Реализуйте объект – итератор списка с методами Init, Done и Next, выдающий при каждом обращении к методу Next очередной элемент списка от начала к концу, и с его помощью просуммируйте элементы списка.

8. Опишите и реализуйте итератор списка, выдающий элементы от его начала к концу, в виде методов списка: IterInit, IterDone и IterNext.

9. Опишите и реализуйте итератор односвязного циклического списка, выдающий элементы от его начала к концу, в виде методов списка: IterInit, IterDone и IterNext.

10. Опишите и реализуйте итератор двусвязного циклического списка, выдающий элементы от его начала к концу, в виде методов списка: IterInit, IterDone и IterNext.

Бинарные деревья. Деревья поиска

1. Дана последовательность ключей 1, 2, 3, 4, 5, 6, 7, 8, 9. Постройте бинарное дерево минимальной высоты с такими ключами.

2. Постройте дерево выражения по постфиксной записи $ab \div de \div c \div$.

3. Постройте дерево выражения по префиксной записи $\div \div ab \div \div dec$.

4. Задано AVL-дерево 54(42, 67(63, 84)). В это дерево последовательно добавляйте ключи 93 и 90, показывая на каждом шаге вид дерева, значения показателей сбалансированности его вершин и вид балансировки.

5. Задано AVL-дерево 18(10(8, 14), 30). В это дерево последовательно добавляйте ключи 4 и 7, показывая на каждом шаге вид дерева, значения показателей

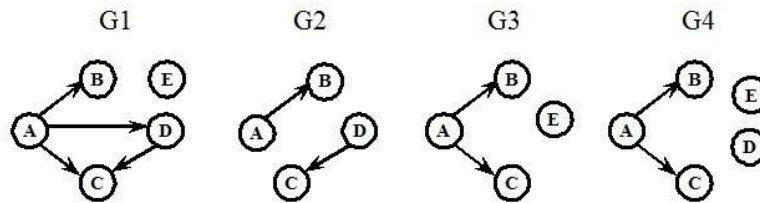
сбалансированности его вершины вид балансировки.

6. Существуют ли два AVL-дерева, у которых высота одинакова (не больше 10), а число вершин отличается на 800.

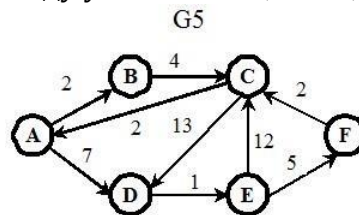
7. Считаем, что пустое дерево имеет высоту 0, дерево из одной вершины – высоту 1. Какова наибольшая высота у AVL-деревьев, содержащих 145 вершин? Обосновать ответ.

Графы. Алгоритмы на графах

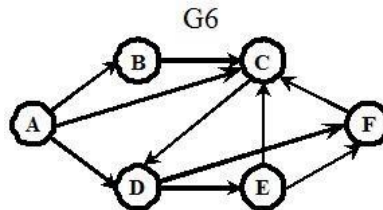
1. Чем являются графы G2, G3, G4 для графа G1?



2. Нарисуйте полный граф с узлами графа G1.
3. Нарисуйте дополнение графа G1.
4. Представьте граф G1 с помощью линейных односвязных списков.
5. Вычислите расстояния между узлами A и F, C и D, B и F, C и F в графе G5.



6. Является ли граф G5 связным?
7. Является ли граф G5 сетью?
8. Составьте матрицы инцидентности и смежности для графа G6.



9. Запишите множества узлов и дуг графа G6 в формате, необходимом для его построения.
10. Опишите последовательность вычисления расстояния между узлами A и F графа G6 и приведите список узлов, входящих в это расстояние.

Сортировка

1. Выполните упорядочивание последовательности 1, 7, 3, 2, 0, 5, 0, 8 с помощью методов «пузырька», сортировки вставками, сортировки посредством выбора.
2. Выполните упорядочивание последовательности 22, 36, 6, 79, 26, 45, 75, 13, 31, 62, 27, 76, 33, 16 используя быструю, пирамидальную сортировки, сортировку вставками.
3. Предположим, что необходимо сортировать список элементов, состоящий из уже упорядоченного списка, который следует за несколькими «случайными» элементами. Какой из рассмотренных методов сортировки наиболее подходит для решения такой задачи?
4. Какой из рассмотренных методов сортировки устойчив?
5. Какие методы сортировки обладают естественным поведением?
6. Предположим, что в алгоритме быстрой сортировки в качестве опорного элемента выбирается первый элемент сортируемого подмножества. Какие изменения следует сделать в алгоритме быстрой сортировки, чтобы избежать «зацикливания» в случае последовательности равных элементов?
7. Рассмотрим алгоритм *случайной сортировки*, примененный к массиву $A[1..n]$ целых чисел: выбирается случайное число i из интервала от 1 до n и меняются местами

элементы $A[1]$ и $A[i]$, процесс повторяется до тех пор, пока не будут упорядочены все элементы массива. Каково ожидаемое время выполнения этой сортировки?

8. Напишите программу нахождения *моды* (наиболее часто встречаемого элемента) в последовательности из n элементов.

9. Напишите программу нахождения k наименьших элементов в массиве длиной n . Для каких значений k эффективнее сначала выполнить сортировку всего массива, а затем взять k наименьших элементов, вместо поиска наименьших элементов в неупорядоченном массиве?

10. Напишите программу нахождения наибольшего и наименьшего элементов в массиве.

Может ли эта программа обойтись менее чем $2n + 3$ сравнениями?

11. Напишите программу вычисления *медианы* последовательности из n элементов (элемент, значение которого больше либо равно значений половины элементов и меньше либо равно значений другой половины элементов).

Поиск и кодирование данных

1. Оцените минимальное и максимальное количество сравнений в алгоритме поиска с бинарным. За счёт чего достигается уменьшение числа операций по сравнению с линейным поиском?

2. В каких случаях целесообразно использовать для поиска методы с переупорядочиванием таблицы. Сравните по эффективности методы перемещения в начало и транспозиции.

3. Оцените минимальное и максимальное количество сравнений в алгоритме бинарного поиска.

4. Каково оптимальное соотношение между размерами таблицы и индекса в индексно-последовательном поиске.

5. Дана таблица $T[0..12]$ с функциями первичной расстановки $i \mapsto k \bmod 13$ и

повторной расстановки $i \mapsto (i \mapsto 4) \bmod 13$, где i – адрес, k – ключ. Указать последовательность

поступления ключей в таблицу. Построить дерево поиска из этих же ключей, поступающих в том же порядке?

i	0	1	2	3	4	5	6	7	8	9	10	11	12
$T[i]$		27				6	19			18	23		

6. Даны две таблицы расстановки $T_1[0..12]$ и $T_2[0..12]$ с функциями первичной расстановки $i \mapsto k \bmod 13$ и повторной расстановки $i \mapsto (i \mapsto 4) \bmod 13$, где i – адрес, k – ключ.

Дополнить

таблицу T_2 ключами из T_1 , чтобы порядок записи ключей был таким же, как и при заполнении

T_1 .

i	0	1	2	3	4	5	6	7	8	9	10	11	12
$T_1[i]$		23				27	19		59				
$T_2[i]$		40				14				5	36		

7. Даны две таблицы расстановки $T_1[0..10]$ и $T_2[0..10]$ с функциями первичной расстановки $i \mapsto k \bmod 11$ и повторной расстановки $i \mapsto (i \mapsto 3) \bmod 11$, где i – адрес, k – ключ.

Дополнить

таблицу T_2 ключами из T_1 , чтобы порядок записи ключей был таким же, как и при заполнении

T_1 .

i	0	1	2	3	4	5	6	7	8	9	10
$T_1[i]$	27	12			23	16		45	19		
$T_2[i]$	11			35				18			

8. Дана таблица $T[0..16]$ с функциями первичной расстановки $i \mapsto k \bmod 17$ и повторной расстановки $i \mapsto (i \cdot 5) \bmod 17$, где i – адрес, k – ключ. Как выглядела бы таблица, если бы эти же ключи заносились в неё в обратном порядке?

i	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
$T[i]$		35			38			24	59					42	48		

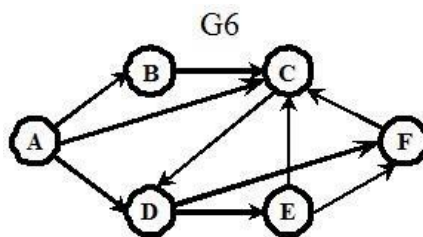
Теория сложности алгоритмов

1. Что такое трудоемкость алгоритма и от чего она зависит?
2. Что называется функцией трудоемкости и как определяется операционная сложность алгоритма?
3. Какие виды сложности алгоритма Вам известны?

4. Сколько видов асимптотик принято использовать в асимптотическом анализе алго-ритмов и как обозначаются асимптотики различных видов?
5. Каково формальное определение асимптотически точной оценки функции трудоем-кости?
6. Каковы формальные определения асимптотически точных верхней и нижней оценок функции трудоемкости?
7. В каких случаях возможно совпадение верхней и нижней оценок трудоемкости алго-ритма?
8. Каковы правила оценки порядка операционной сложности?
9. Каковы асимптотические оценки функции трудоемкости изученных ранее алгорит- мов поиска и сортировки для массива размером n (последовательный и бинарный поиск, сортировка методом простого обмена)?
10. Каковы отличия массовых и индивидуальных задач?
11. Какие задачи называются алгоритмически разрешимыми?
12. Что является целью теоретического исследования задачи?
13. Какие процедуры используются для решения алгоритмически не разрешимых задач?
14. Как определяется сложность задачи и на какие группы делятся алгоритмически раз-решимые задачи?
15. Какова классификация распознавательных задач?
16. Каково отношение между задачами классов **P**, **NP**, **NPC**?
17. Каковы особенности задач, относящихся к классам **P**, **NP**, **NPC**?
18. Каковы асимптотические оценки сложности задачи о выполнимости схемы?
19. Каковы особенности и возможные интерпретации недетерминированной машиныТьюринга?
20. Каковы особенности **NP**-трудных задач?
21. Каким образом можно графически изобразить схему соотношения степеней сложно-сти задач классов **NP** и **NP^H**?
22. В чем состоит суть проблемы **P** $\stackrel{?}{\neq}$ **NP**?

4.2. Типовые контрольные задания итогового контроля

1. С использованием стандартного набора методов составьте программу записи элементовочере-ди в новую очередь в обратном порядке.
2. С использованием стандартного набора методов составьте программу переноса из оче-реди строк в новую очередь элементов, начинающихся на буквы «F» или «f».
3. Реализуйте метод поиска первого и последнего элементов с заданным значением в дву-связном списке.
4. Реализуйте класс – итератор двусвязного списка, выдающий элементы списка от конца к началу, с методами Init, Done и Next.
5. Для заданных последовательностей ключей постройте AVL-дерево, показывая на каж- дом шаге вид дерева, значения показателей сбалансированности его вершин и вид балансиров- ки:
 - 1) 1, 2, 3, 4, 5, 6; 2) 1, 2, 13, 14, 25, 36; 3) 4, 5, 6, 3, 1;
 - 4) 21, 12, 11, 14, 25, 36; 5) 4, 5, 7, 2, 1, 3, 6.
6. Считаем, что пустое дерево имеет высоту 0, дерево из одной вершины – высоту 1. Ка-кова наибольшая высота у AVL-деревьев, содержащих 145 вершин? Обосновать ответ.
7. Опишите последовательность обхода графа G6 в глубину и приведите его результаты.



8. Опишите последовательность обхода графа G6 в ширину, приведите его результаты.

9. Дана таблица $T[0..12]$ с функциями первичной расстановки $i \mapsto k \bmod 13$ и

повторной расстановки $i \mapsto (i \mapsto 5) \bmod 13$, где i – адрес, k – ключ. Указать последовательность

поступления ключей в таблицу. Построить дерево поиска из этих же ключей, поступающих в том же порядке?

i	0	1	2	3	4	5	6	7	8	9	10	11	12
$T[i]$	42	40		45			32		66			27	

10. Определить, может ли таблица $T[0..10]$ быть таблицей расстановки при функциях

первичной расстановки $i \mapsto k \bmod 11$ и повторной расстановки $i \mapsto (i \mapsto 2) \bmod 11$, где i – адрес, k –

ключ. Если «да», то указать последовательность поступления ключей в таблицу. Если «нет», то доказать это.

i	0	1	2	3	4	5	6	7	8	9	10
$T[i]$	20		22		29		31	18	17	7	

4.3. Вопросы к экзамену по дисциплине

1. Логическая и физическая структуры данных. Классификация структур данных по различным признакам. Типовые операции над структурами данных.
2. Стек. Определение. Логическая структура. Операции над стеком.
3. Связанное представление стека, описание класса «стек», процедуры реализации операций. Использование стека для преобразования выражений.
4. Очередь. Определение. Логическая структура. Операции над очередью.
5. Связанное представление очереди, описание класса «очередь», процедуры реализации операций.
6. Дек. Определение. Логическая структура. Операции над деком.
7. Связанное представление дека, описание класса «дек», процедуры реализации операций.
8. Линейный односвязный список. Логическая структура. Операции над линейным списком. Описание класса «линейный список», процедуры реализации операций.
9. Циклический односвязный список. Логическая структура. Операции над циклическим списком. Описание класса «циклический список», процедуры реализации операций.
10. Линейный двусвязный список. Операции над линейным двусвязным списком. Описание класса «линейный двусвязный список», процедуры реализации операций.
11. Циклический двусвязный список. Операции над циклическим двусвязным списком. Описание класса «циклический двусвязный список», процедуры реализации

операций.

12. Деревья. Определение. Изображение. Терминология.
13. Бинарные деревья. Определение. Терминология. Операции над бинарным деревом.
14. Реализация бинарного дерева. Описание класса «бинарное дерево».
15. Рекурсивная процедура создания бинарного дерева минимальной высоты.
16. Рекурсивная процедура печати узлов бинарного дерева.
17. Примитивные операции над бинарными деревьями. Реализация операций «адрес отца вершины с адресом Addr», «удаление всех узлов дерева».
18. Обход бинарного дерева. Способы и рекурсивные процедуры обхода.

19. Дерево поиска. Терминология. Операции над деревом поиска.
20. Описание класса «дерево поиска».
21. Процедура поиска элемента с заданным ключом.
22. Процедура построения дерева поиска.
23. Реализация рекурсивной процедуры поиска с исключением.
24. Разнородные бинарные деревья. Рекурсивный алгоритм и процедура построения дерева выражения.
25. Сбалансированные АВЛ-деревья. Включение в сбалансированное дерево (LR- и LL-поворот).
26. Графы. Определение. Изображение. Терминология. Операции над графами.
27. Представление графов различных видов (ориентированных и неориентированных, взвешенных и невзвешенных) с использованием линейных односвязных списков.
28. Описание класса «ориентированный невзвешенный граф» с использованием линейных односвязных списков.
29. Процедуры реализации основных операций над графом.
30. Процедура вычисления расстояния между узлами графа без циклов.
31. Процедура построения всех путей между двумя узлами графа с циклами и петлями.
32. Поиск информации. Терминология. Оценка эффективности алгоритмов поиска.
33. Последовательный поиск в таблице, организованной в виде массива и списка.
- Поиск с включением. Поиск с барьером.
34. Поиск с переупорядочением списка (перемещение в начало, транспозиция).
- Сравнение методов переупорядочивания.
35. Поиск в упорядоченной таблице. Индексно-последовательный поиск.
36. Бинарный поиск. Характеристики методов.
37. Преобразование ключей (расстановка). Основные определения. Сравнение различных методов разрешения конфликтов при расстановке.
38. Функции расстановки (деления, середины квадрата, свертки, преобразование системы счисления, алгебраическое кодирование).
39. Разрешение конфликтов при расстановке методом линейного опробования.
- Программа поиска. Достоинства и недостатки метода.
40. Разрешение конфликтов при расстановке методом случайного опробования.
- Двойная расстановка. Характеристики метода.
41. Разрешение конфликтов методом раздельного сцепления.
42. Разрешение конфликтов методом внутреннего сцепления.
43. Внутренняя сортировка. Терминология. Оценка эффективности.
44. Сортировка подсчетом. Процедура сортировки.
45. Сортировка простыми вставками. Процедура сортировки.
46. Сортировка бинарными вставками. Процедура сортировки.
47. Сортировка вставками в список. Процедура сортировки.
48. Сортировка с вычислением адреса. Процедура сортировки.
49. Обменная сортировка. Простой обмен. Процедура сортировки.
50. Обменная сортировка. Процедура сортировки.
51. Обменная сортировка. Шейкер-сортировка. Процедура сортировки.
52. Обменная сортировка. Метод четных и нечетных транспозиций.
53. Быстрая сортировка. Процедура сортировки.
54. Сортировка простым выбором. Процедура сортировки.
55. Понятие трудоемкости алгоритма.
56. Сложность алгоритма. Виды и обозначения асимптотик.
57. Оценивание средней трудоемкости и операционной сложности алгоритма.

- 58. Классификация распознавательных задач.
- 59. Недетерминированная машина Тьюринга.
- 60. **NP** -трудные задачи.

Основная учебная литература

1. Белов В.В., Чистякова В.И. Алгоритмы и структуры данных: Учебник. – М.: КУРС: ИНФРА-М, 2016. – 240 с. (100 экз. в БФ РГРТУ).
2. Белов В.В., Чистякова В.И. Программирование в Delphi: процедурное, объектно-ориентированное, визуальное. 3-е изд. Учебное пособие для вузов. – М.: Горячая линия–Телеком, 2017. – 240 с. (56 экз. в БФ РГРТУ).
3. Никлаус Вирт Алгоритмы и структуры данных [Электронный ресурс]/ Никлаус Вирт— Электрон. текстовые данные.— Саратов: Профобразование, 2017.— 272 с.— Режим доступа: <http://www.iprbookshop.ru/63821.html>.— ЭБС «IPRbooks»

Дополнительная литература

4. Бакнелл Джулиан М. Фундаментальные алгоритмы и структуры данных в Delphi: Пер.с англ./ Джулиан М. Банкнелл.– СПб: ООО «ДиаСофт ЮП», 2013. – 360 с. Доступно по URL: <https://www.twirpx.com/file/835015>, <https://royallib.com/book>.
5. Ахо Альфред В. Структуры данных и алгоритмы: Пер. с англ./ Альфред В. Ахо, Джон Э. Хопкрофт, Д. Ульман Джеффри. – Москва: Изд. дом «Вильямс», 2010. – 384 с. Доступно по URL: <http://knigovodstvo.ru/book/1526>.

5. Перечень ресурсов информационно-телекоммуникационной сети

Интернет для самостоятельной работы

1. Электронно-библиотечная система «Лань». – Режим доступа: с любого компьютера РГРТУ без пароля. – URL: <https://e.lanbook.com/>
2. Электронно-библиотечная система «IPRbooks». – Режим доступа: с любого компьютера РГРТУ без пароля, из сети Интернет по паролю. – URL: <https://iprbookshop.ru/>.
3. Электронная библиотека РГРТУ. – URL: <http://weblib.rtu/ebs>.
4. Научная электронная библиотека eLibrary.
– URL: <http://e.lib/vlsu.ru/www.uisrussia.msu.ru/elibrary.ru>
5. Библиотека и форум по программированию. – URL: <http://www.cyberforum.ru>
6. Национальный открытый университет ИНТУИТ. – URL: <http://www.intuit.ru/>
7. Информационно-справочная система. – URL: <http://window.edu.ru>

Оценочные материалы составлены в соответствии с рабочей программой дисциплины «Алгоритмы и структуры данных» по специальности 09.05.01 «Применение и эксплуатация систем специального назначения» (уровень специалитета).

Составил:

к.т.н., доцент кафедры
«Вычислительная и прикладная математика»

С.Ю.Жулева