

# Программа экзамена "Работа на ЭВМ и программирование"

## 2 курс, лектор В.Д.Валединский 2022–2023 учебный год.

1. Основные конструкции языка C++ (см. отдельный список).
2. Динамический массив. Идеи реализации.
3. Стек, дек, очередь. Непрерывные реализации на базе массива.
4. Ссылочные реализации списков (одно- и двунаправленного).
5. Бинарное дерево поиска. Реализация; процедуры добавления, удаления, обхода.
6. Произвольное дерево. Реализация; процедуры добавления, удаления, обхода.
7. Сбалансированное (AVL) бинарное дерево; процедуры добавления, удаления. Теорема о глубине AVL дерева.
8. Красно-черное дерево; процедуры добавления, удаления. Теорема о глубине красно-черного дерева.
9. B-дерево, B+ дерево, 2-3 дерево; процедуры добавления, удаления. Трудоемкость операций с B-деревом.
10. Итераторы. Итератор по списку. Итератор по дереву с использованием стека. Реализация как набора методов, как отдельного внутреннего или внешнего классов.
11. Реализации графов. Поиск в ширину и в глубину. Сложности поисков. Обнаружение циклов.
12. Алгоритм Дейкстры для поиска кратчайшего пути между двумя вершинами в графе. Сложность поиска.
13. Битовая реализация множества.
14. Хеширование. Примеры хеш-функций. Хеш-множество на базе массива списков.
15. Хеш-множество по методу последовательных проб. Оценка среднего количества проб при поиске и добавлении.
16. Совершенная (perfect) хеш-функция; определения, пример построения.
17. Контейнеры для элементов фиксированного и произвольного размера. Идеи реализации управления памятью в общем случае (функции malloc и free).
18. Теорема об оценке снизу для трудоемкости сортировки сравнениями.
19. Быстрая сортировка (quicksort), оценка средней трудоемкости.
20. Алгоритм группового кодирования (RLE), байтовый и битовый варианты.
21. Алгоритм Хаффмена (обычный и адаптивный). Теорема об оптимальности кода Хаффмена.
22. Арифметическое кодирование (обычное и адаптивное). Теорема о сжатии при арифметическом кодировании.
23. Алгоритм LZW. Процедуры кодирования и декодирования, увеличение длины кода по мере разрастания словаря.

### Форма проведения экзамена:

1. Письменный ответ на билет с вопросами, относящимися к темам представленной выше программы.
2. Практическая задача на реализацию некоторых алгоритмов (или их частей) из лекционного курса.

### Примерная структура письменного билета:

1. Вопрос по базовым понятиям языка C++.
- 2-3. Определение понятия, формулировка утверждения.
4. Вопросы о сути алгоритмов, доказательств, схем представлений данных.
- 5-6. Задачи по выполнению или оценке свойств алгоритмов на модельном наборе данных.

### Содержание практической задачи:

Даны некоторые заготовки, создающие некоторые структуры данных с заданными свойствами, например, деревья или графы. Требуется выполнить обход или поиск в этих структурах с целью определения или вычисления некоторых характеристик записанных там наборов данных. В зависимости от формулировки задачи, обход может выполняться рекурсивной процедурой, либо должен быть реализован с помощью итератора заданного вида (который также нужно реализовать).

Как пример задания — найти в бинарном дереве все вершины, нарушающие условие AVL сбалансированности или проверить обладает ли данное дерево указанным свойством (например, является ли деревом Хаффмена), также возможны задачи на небольшие преобразования, например, перестановка поддеревьев, по некоторому условию, обнаружение поддеревьев, обладающих схожей структурой, поиск путей в деревьях и графах и т.п.

На выполнение письменной части отводится 45-50 минут, на реализацию программ — 90 минут.

### Пример письменного билета:

1. C++: Когда и для чего используется слово static ?
2. Что такое итератор?
3. Формулировка теоремы о глубине AVL дерева.
4. Опишите кратко алгоритм поиска циклов в неориентированном графе.
5. В красно-черное дерево последовательно добавляются элементы 1, 2, 8, 10, 12, 20, а потом удаляется элемент 8. Изобразите последовательные состояния дерева после выполнения каждого добавления и удаления.
6. Раскодируйте последовательность 001110101111000101 адаптивным методом Хаффмена для алфавита  $\{a, b, c, d\}$ .

### Критерии оценивания:

Полностью выполненное задание семестра дает 2 балла.

Программа решения практической задачи на экзамене должна выдавать правильный ответ и нормально завершаться (без "падения" и потерь памяти "memory leak"). Если она реализует правильное решение, то дает 1.5 балла. Вопросы теоретического билета оцениваются в  $1/3 \approx 0.33$  балла при правильном и четком ответе на поставленный вопрос. Верные факты и рассуждения, не имеющие прямого отношения к заданным вопросам, правильным ответом не считаются и соответственно снижают оценку ответа на вопрос. При погрешностях в ответах могут быть выставлены промежуточные оценки  $\pm, \mp$  соответственно в  $2/9$  и  $1/9$  балла.

Поскольку максимально возможная сумма баллов составляет 5.5, то округление, как правило, проводится в сторону уменьшения, т.е. 4.5 — это 4, а 3.5 — это 3.

**Исключения:**

Только семестровое задание и решенная практическая задача на экзамене (2+1.5) не достаточны сами по себе для получения оценки 4. Здесь должен присутствовать ответ на теоретический билет, превышающий 0.5 балла (т.е., например, два +).

Пустое семестровое задание (0 баллов) при полном ответе на экзамене (1.5 + 2) также дает итоговую оценку в 3 балла.