

Список вопросов к экзамену/зачету по курсу
«Информатика (организация и поиск данных)»

1. Структуры данных
 - 1.1. Классификация структур данных
 - 1.2. Массивы, динамические массивы. Основные операции, асимптотика.
 - 1.3. Связанные списки и их вариации: односвязные и двусвязные, линейные и кольцевые. Основные операции, асимптотика.
 - 1.4. Очередь. Способы реализации и оценка асимптотики.
 - 1.5. Стек. Способы реализации и оценка асимптотики.
 - 1.6. Двоичная куча. Основные операции. Способы реализации и оценка асимптотики.
 - 1.7. Поток данных (stream). Основные операции, способы реализации, сравнительная характеристика и асимптотика.
 - 1.8. Очередь с приоритетами. Способы реализации и оценка асимптотики.
 - 1.9. Бинарное дерево поиска. Основные операции. Балансировка. Оценка асимптотики.
 - 1.10. Понятие итератора, основные операции
 - 1.11. Тип Option<>
 - 1.12. Тип Either<>
 - 1.13. N-арные деревья. B- и B+-деревья. Основные операции, оценка асимптотики.
 - 1.14. Хеш-таблицы. Хеш-функции. Разрешение коллизий в хеш-таблицах.
 - 1.15. Общие алгоритмы на перечислимых контейнерах: map, reduce, where.
 - 1.16. Понятие монады (как идиомы и структуры данных). Пример
2. Алгоритмы сортировки
 - 2.1. Алгоритмы сортировки: классификация
 - 2.2. Обменные сортировки: общий принцип, примеры, асимптотическая оценка сложности
 - 2.3. Сортировки выбором: общий принцип, примеры, асимптотическая оценка сложности
 - 2.4. Сортировки вставкой: общий принцип, примеры, асимптотическая оценка сложности
 - 2.5. Декомпозиция задач на примере алгоритмы быстрой сортировки
3. Ленивые вычисления
 - 3.1. Концепция ленивой последовательности. Логическое определение и внешняя спецификация
 - 3.2. Концепция ленивой последовательности. Основные приемы реализации
 - 3.3. Понятие ординального числа. Использование ординальных чисел для индексирования последовательностей
 - 3.4. Арифметика ленивых последовательностей. Конкатенация, вставка. Удаление подпоследовательности.
 - 3.5. «Двунаправленная» ленивая последовательность. Приемы реализации
 - 3.6. Поток данных (stream). Назначение и принцип реализации
 - 3.7. Поток данных (stream). Пример реализации потока, считывающего данные из бинарного файла
 - 3.8. Поток данных (stream). Пример реализации потока, считывающего данные из массива
 - 3.9. Кодирование потока данных, пример алгоритма
4. Методы поиска

- 4.1. Ассоциативная память, словарь. Варианты реализации: с помощью хеш-таблиц, деревьев поиска. Сравнение асимптотических оценок характеристик алгоритмов (вставка, поиск по ключу).
- 4.2. Полиморфный АТД «Множество»: реализация на базе хеш-таблиц. Асимптотическая оценка сложности вставки элемента и поиска (проверки на вхождение в множество).
- 4.3. Полиморфный АТД «Множество»: реализация на базе дерева поиска (бинарного или В-дерева) . Асимптотическая оценка сложности вставки элемента и поиска (проверки на вхождение в множество).
- 4.4. Организация пространства состояний или решений в виде n-арного дерева: основные идеи (на примере задачи о рюкзаке)
- 4.5. Организация пространства состояний или решений в виде n-арного дерева: основные идеи (на примере минимаксного алгоритма)
- 4.6. Ускорение поиска данных с помощью кэширования.
- 4.7. Ускорение поиска данных с помощью индексирования.
- 4.8. Сравнительная характеристика систем: реального времени, онлайн, асинхронных.
- 4.9. Разреженные структуры: одномерные. Реализация с помощью дерева поиска и с помощью словаря. Сравнительная характеристика, асимптотика.
- 4.10. Разреженные структуры: многомерные. Реализация с помощью ассоциативной памяти. Асимптотика.
- 4.11. Пространственные данные: одномерный случай. Дерево отрезков.
5. Графовые алгоритмы
 - 5.1. Структуры данных для представления ориентированного графа с взвешенными дугами
 - 5.2. Структуры данных для представления неориентированного графа с взвешенными ребрами
 - 5.3. Способы реализации матрицы смежности. Обзор и сравнительная характеристика
 - 5.4. Обход ориентированного графа в ширину (алгоритм)
 - 5.5. Обход ориентированного графа в глубину (алгоритм)
 - 5.6. Обход неориентированного графа в ширину (алгоритм)
 - 5.7. Обход неориентированного графа в глубину (алгоритм)
 - 5.8. Полиморфный АТД «Очередь с приоритетами»: реализация на основе бинарной кучи, основные операции, асимптотическая оценка сложности
 - 5.9. Полиморфный АТД «Очередь с приоритетами»: реализация на основе связанного списка, основные операции, асимптотическая оценка сложности
 - 5.10. Поиск кратчайших путей: алгоритм Дейкстры
 - 5.11. Динамическое программирование: нисходящий процесс
 - 5.12. Динамическое программирование: восходящий процесс
 - 5.13. Раскраска графа. Алгоритм Ершова
 - 5.14. Поиск остова графа. Алгоритм Прима
 - 5.15. Поиск остова графа. Алгоритм Краскала
6. Идиомы и приемы объектно-ориентированного программирования
 - 6.1. Указатель как идиома и варианты реализации через индексирование коллекций и с помощью итераторов
 - 6.2. Арифметика указателей в C/C++
 - 6.3. Переопределение операторов в C++
 - 6.4. Особенности переопределения операторов new и delete

- 6.5. Умные указатели
- 6.6. out-параметры в функциях и методах
- 6.7. Шаблоны и параметрический полиморфизм. Связь с подтипизацией.
- 6.8. Наследование, адгос-полиморфизм и правила переопределения методов и их перегрузок
- 6.9. Изменяемые и неизменяемые структуры: сравнительная характеристика.
- 6.10. Объекты как глобальные константы: ленивая реализация с помощью мемоизации
- 6.11. Контроль и управление памятью: копирующие конструкторы, конструкторы перемещения, l- и r-значения, виртуальные и неvirtуальные деструкторы, порядок инициализации и уничтожения объектов
- 6.12. Абстрагирование зависимостей и разделение ответственностей (на примере Sequence)
- 6.13. Инкапсуляция и изоляция. Связь с friend-классами.
- 6.14. Понятие объектной модели. Пример
- 6.15. Понятие атрибута объекта. Реализация концепции атрибута средствами ООП
- 6.16. Подтипизация и наследование. Изменение внешней спецификации при подтипизации
- 6.17. Подтипизация и наследование. Проблемы множественного наследования. Ромбовидное наследование

Экзамен и зачет сдаются в устной форме, по билетам. Билеты содержат 2 вопроса из разных разделов. Ответ на каждый вопрос оценивается по шкале 0-5 баллов. Оценки за оба вопроса суммируются. Полученная величина суммируется с оценками за разделы, эта сумма и составляет итоговую оценку по курсу.