

Алгоритмы и структуры данных

Курс читается на английском языке.

Предполагаемый день занятий – понедельник.

Занятия 1 лекция и 1 семинар в неделю.

От слушателей требуется знакомство с основами программирования на языке C.

Лекция 1. Сложность и модели вычислений. Анализ учетных стоимостей (начало)

Основные ресурсы: память и время. O-символика. Примеры моделей вычисления: машина Тьюринга, RAM-машина. Сложность в среднем и худшем случаях. Пример: задача сортировки. Сортировка выбором. Теоретико-информационная нижняя оценка сложности. Разрешающие деревья. Нижняя оценка сложности в модели разрешающих деревьев. Массивы переменного размера: аддитивная и мультипликативная схемы реаллокации. Анализ мультипликативной схемы для массива переменного размера с помощью банковского метода.

Лекция 2. Анализ учетных стоимостей (окончание)

Анализ учетных стоимостей операций: функция потенциала, истинные и учетные стоимости. Стеки и очереди. Реализация на основе массива переменного размера и на основе связанного списка. Моделирование очереди с помощью двух стеков. Задача о поддержании динамического максимума в стеке и очереди. Изменяемые (mutable) и неизменяемые (immutable) структуры данных. Структуры данных с хранением истории (persistent). Immutable-стек и immutable-очередь. Проблема множественного будущего при анализе учетных стоимостей в persistent-структурах.

Лекция 3. Алгоритмы Merge-Sort и Quick-Sort

Понятие о методе «разделяй и властвуй». Алгоритм Merge-Sort. Слияние двух упорядоченных списков. Оценка сложности. K-way Merge-Sort для работы во внешней памяти. Сортировка слиянием без использования дополнительной памяти. Общая схема алгоритма Quick-Sort. Два варианта реализации Partition. Примеры неудачного выбора опорных элементов. Рандомизированный выбор опорного элемента. Сложность Quick-Sort в худшем и среднем случаях. Глубина рекурсии в худшем и среднем случаях. Элиминация хвостовой рекурсии. Задача об оптимальном дереве слияний. Коды Хаффмана. Слияние двух упорядоченных последовательностей различной длины. Теоретико-информационная нижняя оценка. Бинарный поиск "от края" (galloping).

Лекция 4. Порядковые статистики. Кучи (начало)

Нахождение порядковых статистик с помощью рандомизированной модификации алгоритма Quick-Sort. Линейность матожидания времени работы. Приближенные медианы. Выбор k-й порядковой статистики за линейное в худшем случае. Деревья со свойствами кучи. Почти полные бинарные деревья: нумерация вершин, навигация. Двоичная куча. Операция просеивания вниз и вверх. Реализация операций вставки, удаления и поиска минимума. Преобразование произвольного массива ключей в кучу (операция Make-Heap), линейность времени работы. Алгоритм сортировки Heap-Sort.

Лекция 5. Кучи (окончание)

к-ичные кучи, зависимость сложности операций от выбора k . Биномиальные (binomial), левацкие (leftlist) и косые (skew) кучи.

Лекция 6. Хеширование

Хеш-функции. Коллизии. Разрешение коллизий методом цепочек, методом последовательных проб и методом двойного хеширования. Гипотеза простого равномерного хеширования, оценка средней длины цепочки. Универсальные семейства хеш-функций, оценка средней длины цепочки. Построение универсального семейства для целочисленных ключей. Совершенные хеш-функции. Построение совершенной хеш-функции с помощью универсального семейства. Интерфейс множества с ошибками. Фильтр Блума (Bloom filter). Оценка вероятности ложноположительного срабатывания. Интерфейс словаря с ошибками. Модификация фильтра Блума (bloomier filter).

Лекция 7. Деревья поиска (начало)

Определение дерева поиска. Вставка и удаление элементов. Inorder-обход дерева. Красно-черные деревья: определение и основные свойства. Реализация операций вставки для красно-черного дерева. Splay-деревья. Операция splay: zig, zig-zig и zig-zag шаги. Реализация операций вставки, удаления, слияния и разделения для splay-деревьев.

Лекция 8. Деревья поиска (продолжение)

Декартовы деревья (дучи). Единственность декартова дерева для заданного набора различных ключей и приоритетов. Логарифмическая оценка матожидания высоты дучи. Операции слияния и разделения для дуч. Операции вставки и удаления элементов для дуч. Построение декартового дерева за линейное время при условии предварительной сортировки ключей.

Лекция 9. Деревья поиска (окончание). Система непересекающихся множеств

B^+ деревья: определения и основные свойства. Операции поиска, вставки и удаления для B^+ деревьев. Системы непересекающихся множеств. Реализация с использованием леса. Ранги вершин, эвристика ранга. Логарифмическая оценка ранга через количество элементов. Рандомизированная ранговая эвристика. Эвристика сжатия путей. Оценка учетной стоимости операций (без доказательства).

Лекция 10. Задачи RMQ и LCA

Задачи RMQ (range minimum query) и LCA (least common ancestor). Сведение от задачи RMQ к задаче LCA, декартово дерево. Алгоритм Таржана для offline-версии задачи LCA. Простейшие алгоритмы для online-версии задачи LCA: полная и разреженная таблицы ответов. Алгоритм Фарах-Колтона-Бендера для k задаче ± 1 -RMQ. Сведение задачи LCA к задаче ± 1 -RMQ: эйлеров обход дерева.

Лекция 11. Структуры данных для геометрического поиска

Location problem, stabbing problem. Деревья интервалов. Сведение системы интервалов к двумерной задаче. Задача поиска точек в коридоре. Priority search tree. Задача поиска точек в прямоугольнике. Дерево отрезков по координате X , упорядоченные по Y списки точек в каждой вершине. Сложность $O(n \log n)$ для построения и $O(\log^2 n)$ для запроса. Уменьшение времени поиска до $O(\log n)$. Задача одновременного поиска в наборе упорядоченных списков. Fractional cascading.

Лекция 12. Задача о динамической связности в ненаправленном графе

Задача о динамической связности: вставки и удаления ребер, запросы о связности. Частный случай задачи для случая лесов. Дервья эйлеровых обходов: слияние и разделение. Использование амортизации и набора остовных лесов для решения со сложностью $O(\log^2 n)$.