

Вопросы к экзамену по алгоритмам SPb HSE, 1 курс, июнь 2021

Общая информация

- Кроме конспектов полезно смотреть **разборы** задач из практик.
- *Курсивом* помечено то, что было разобрано на практике.
- (a) темы на 3 (оценка 4-5).
- (b) темы на 4 (оценка 6-7).
- (c) темы на 5 (оценка 8-9).
- (+) факультативные темы (оценка 10) – нужно заботать 2 любых темы из 6.

Бинарные деревья поиска

- (a) 1. BST. Add, Del, Find, LowerBound за $\mathcal{O}(h)$. Симметричный обход, сортировка, нижняя оценка.
- (a) 2. BST. Next, Prev, Find за $\mathcal{O}(1)$. Список. Хеш-таблица. Итератор в `set`.
- (b) 3. BST. Обработка равных ключей. Удаление за $\mathcal{O}(1)$.
- (b) 4. BST. Персистентная версия добавления, удаления.
- (c) 5. BST. *Прямой обход*. Восстановление дерева за $\mathcal{O}(n)$.
- (a) 6. AVL. Инвариант, типы вращений. Оценка глубины. Add.
- (b) 7. AVL. *Del, Merge (без док-ва перебалансировки на k)*. Персистентное вращение.
- (c) 8. AVL. Число вращений при Add и *Del*.
- (c) 9. AVL. *Split (с док-вом перебалансировки на k)*.
- (a) 10. Общие идеи: BST и неявный ключ, BST и персистентность.
- (b) 11. Общие идеи: запрос на отрезке BST-дерева, отложенные операции, разворот отрезка.
- (a) 12. B-дерево. Find, Add. Оценка $\mathcal{O}(k \log n_k)$.
- (b) 13. B-дерево. Find, Add, Del. Оценка $\mathcal{O}(\log n)$. *Split, Merge*.
- (b) 14. B-дерево. Вариации. B*-дерево, B⁺ дерево, 2-3-Tree, RB, AA.
- (a) 15. Треар. Два определения случайного дерева (RBST). Эквивалентность определений. Оценка средней глубины вершины.
- (a) 16. Треар. Единственность дерева при уникальных y , связь со случайными деревьями. Операции Split, Merge.
- (b) 17. Треар. *Эффективная реализация Add, Delete*. Персистентные декартовы деревья.
- (b) 18. Треар. Garbage collector на примере персистентного декартового дерева (ссылочный, стековый).
- (a) 19. Splay. Три вида поворотов. Операции Splay, Add. Формулировка главной теоремы о времени работы.
- (b) 20. Splay. Del, *Split, Merge*. Потенциал. Оценка всего, кроме операции Splay.
- (c) 21. Splay. Доказательство амортизированной оценки операции Splay.
- (c) 22. Splay. Теорема о статической оптимальности. Гипотеза динамической оптимальности.

Структуры данных

- (a) 23. Persistent. Массив – любой способ. Персистентный дек через Pairing.

- (a) 24. Persistent. Offline, дерево версий. *Детская персистентность для массива (2 версии), частичная персистентность за $\mathcal{O}(\log n)$.*
- (b) 25. Persistent. *СНМ*. Персистентный массив: BST vs дерево отрезков.
- (c) 26. Persistent. Стек, очередь (через 5 стеков).
- (b) 27. Rope. Skip-List. Операции Find, Insert, Erase, Split, Merge.
- (a) 28. Корневая декомпозиция. Статичная по массиву (с примером).
- (b) 29. Корневая декомпозиция. Через Split/Merge. Через Split/Rebuild. Примеры.
- (b) 30. Корневая декомпозиция. Оптимальный выбор размера куска на примере задачи (kth_stat + reverse).
- (c) 31. *Корневая декомпозиция по запросам на примере «Dynamic Connectivity Offline за $\mathcal{O}(m\sqrt{m})$ ».*
- (a) 32. Дерево отрезков. Реализация сверху. Оценка на память, на время get.
- (a) 33. Дерево отрезков. Массовые модификации. += и getMin. = и getSum.
- (b) 34. Дерево отрезков. Реализация снизу. Корректность. Сравнение с реализацией сверху.
- (b) 35. Дерево отрезков. Сравнение с BST по неявному ключу.
- (b) 36. Дерево отрезков. Динамическое (два способа), сжатие координат. Применение сжатия координат для задач со scanline.
- (a) 37. Scanline. Число точек в прямоугольниках; число прямоугольников, накрывающих точку.
- (c) 38. Scanline. Лежит ли точка внутри невыпуклого многоугольника?.
- (b) 39. Scanline. *Площадь объединения прямоугольников.*
- (a) 40. 2D. ДО из сортированных массивов. Связь с 2-мерным запросом (количеством точек в прямоугольнике).
- (b) 41. 2D. Количество точек в прямоугольнике online за $\mathcal{O}(\log n)$.
- (b) 42. 2D. Поиск k -й порядковой статистики на отрезке за $\mathcal{O}(\log^2 n)$.
- (c) 43. 2D. Многомерные структуры. ДО из ДО (зачем нужно?), ДО из ДД (зачем нужно?). Трёхмерное ДО, k -мерное ДО.
- (c) 44. 2D. Поиск k -й порядковой на отрезке за $\mathcal{O}(\log n)$.

RMQ, LCA, LA

- (a) 45. RMQ. Разреженные таблицы.
- (b) 46. RMQ. Фенвик.
- (a) 47. RMQ. Сведение LCA \rightarrow RMQ ± 1 . Решение задачи LCA за $\langle \mathcal{O}(n \log n), \mathcal{O}(1) \rangle$.
- (b) 48. RMQ. Модификации разреженных таблиц. $\langle \mathcal{O}(n \log \log n), \mathcal{O}(1) \rangle$ и $\langle \mathcal{O}(n), \mathcal{O}(\log \log n) \rangle$.
- (b) 49. RMQ. *Решение задачи RMQ за $\langle \mathcal{O}(n \log^* n), \mathcal{O}(\log^* n) \rangle$.*
- (b) 50. RMQ. Сведение RMQ \rightarrow LCA. Построение декартового дерева за линейное время.
- (c) 51. RMQ. Фарах-Колтон-Бендер. Метод четырёх русских, решение RMQ и LCA за $\langle \mathcal{O}(n), \mathcal{O}(1) \rangle$.
- (c) 52. RMQ. *Фарах-Колтон-Бендер: подсчёт ровно за $\mathcal{O}(2^k)$.*
- (a) 53. LCA. Двоичные подьёмы. Решение LCA, два способа (с одной двигающейся вершиной и с двумя).
- (b) 54. LCA в offline, алгоритм Тарьяна.
- (a) 55. LA. Решение LA в online и в offline.
- (c) 56. LA. Алгоритм Вишкина (решение LA за $\langle \mathcal{O}(n), \mathcal{O}(\log n) \rangle$).
- (a) 57. *Сумма на пути в дереве за $\langle \mathcal{O}(n), \mathcal{O}(1) \rangle$.*
- (c) 58. *Сумма на пути в меняющемся дереве за $\langle \mathcal{O}(n), \mathcal{O}(\log n) \rangle$.*

Другие древесные алгоритмы и структуры

- (a) 59. Heavy-Light Decomposition. Построение за $\mathcal{O}(n)$.

- (a) 60. Heavy-Light Decomposition. Вычисление функции на пути, функции поддеревя.
- (b) 61. Euler-Tour Tree. Операции Link, Cut, IsConnected.
- (b) 62. Link-Cut. Описание структуры. Операции Expose, MakeRoot, Link, Cut.
- (b) 63. Link-Cut. Потенциал. Оценка времени Expose.
- (c) 64. Link-Cut. Оценка времени MakeRoot, *Link*, *Cut*.
- (c) 65. MST за линейное время в среднем. Собственно алгоритм.
- (c) 66. MST за линейное время в среднем. Лемма про количество «небесполезных рёбер».
- (a) 67. Бор. `map<string,int>`. Время работы для различных способов хранения. Про Splay – только формулировка.
- (a) 68. Бор. `Sort(strings)`.
- (b) 69. Бор. Сжатый бор. Суффиксное дерево. Поиск подстроки в тексте.
- (c) 70. Бор. Суффиксное дерево. Число подстрок. Общая подстрока $k \leq w$ строк за $\mathcal{O}(\sum |s_i|)$.

Игры

- (a) 71. Игры. Ацикличный граф. Решение для симметричной/несимметричной игры.
- (b) 72. Игры. Граф с циклами. Ретроанализ. Реализация за $\mathcal{O}(E)$. Пример цикла без ничей.
- (c) 73. Игры. Длина самой короткой игры. Длина самой длинной игры.
- (a) 74. Гранди. Функция Гранди, вычисление, связь с выигрышностью.
- (a) 75. Гранди. Прямая сумма. Теорема Гранди про ксор без доказательства.
Примеры на тему: Ним, *Игра в спички*, *Скамейки*.
- (b) 76. Гранди. Теорема Гранди про ксор: доказательство.
- (b) 77. Гранди. Максимальное значение функции Гранди.

Бонус

- (+) 78. Дополнительные деревья: ленивое полное дерево, китайское дерево.
- (+) 79. Персистентность через Fat Nodes.
- (+) 80. Техника Fractional Cascading для двумерных запросов и параллельного бинпоиска.
- (+) 81. КД-дерево. Ответы на двумерные запросы (и двумерные модификации!) за $\mathcal{O}(\sqrt{n})$, запрос «перечислить все точки в прямоугольнике за $\mathcal{O}(k + \log n)$ ».
- (+) 82. LA: Вишкин за $\langle \mathcal{O}(n \log n), \mathcal{O}(1) \rangle$, решение LA через ladders decomposition.
- (+) 83. LA: решение за $\langle \mathcal{O}(n), \mathcal{O}(1) \rangle$ через 4 русских.
- (+) 84. Смит (Гранди для графов с циклами). Без доказательства.