

Вопросы к экзамену по алгоритмам SPb HSE, 1-й курс ПМИ, июнь 2025

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 8 **любых** темы из 11.

Модели вычислений

- (a) 1. RAM и RAM-w, определения, различия.
- (a) 2. RAM-w. Флойд за $\mathcal{O}(n^3/w)$ и $\mathcal{O}(n^2)$ в RAM.
- (b) 3. RAM-w. bfs за $\mathcal{O}(n^2/w)$ и $\mathcal{O}(n)$ в RAM.
- (a) 4. RAM. P = NP. Обоснование, общий принцип.
- (b) 5. RAM. Создание числа-массива из n w -битных единиц.
- (b) 6. RAM. Сумма массива из w -битных чисел за $\mathcal{O}(\log n)$.
- (c) 7. RAM. Конкретный алгоритм для 3 – SAT в RAM за $\mathcal{O}(m + n)$.
- (c) 8. Машина Тьюринга. Определение.

Бинарные деревья поиска

- (a) 9. BST. Add, Del, Find, LowerBound за $\mathcal{O}(h)$. Симметричный обход, сортировка, нижняя оценка.
- (a) 10. BST. Next, Prev, Find за $\mathcal{O}(1)$. Список. Хеш-таблица. Итератор в **set**.
- (b) 11. BST. Add/LowerBound оценка снизу $\Omega(\log n)$.
- (b) 12. BST. Обработка равных ключей. Del за $\mathcal{O}(1)$.
- (b) 13. BST. Персистентная версия добавления, удаления.
- (c) 14. BST. *Прямой обход. Восстановление дерева за $\mathcal{O}(n)$.*
- (a) 15. AVL. Инвариант, типы вращений. Оценка глубины. Add.
- (b) 16. AVL. *Del, Merge (без док-ва переконфигурировки на k).* Персистентное вращение.
- (b) 17. AVL. Число вращений при Add и Del.
- (c) 18. AVL. *Split (с док-вом переконфигурировки на k).*
- (a) 19. Общие идеи: BST и неявный ключ, BST и персистентность.
- (b) 20. Общие идеи: запрос на отрезке BST-дерева, отложенные операции, разворот отрезка.
- (a) 21. B-дерево. Find, Add. Оценка $\mathcal{O}(k \log_k n)$ и $\mathcal{O}(\log_k n)$ чтений.
- (b) 22. B-дерево. Find, Add, Del. Оценка $\mathcal{O}(\log n)$. *Split, Merge.*
- (b) 23. RB-дерево. Определение.
- (c) 24. B-дерево. Вариации. B*-дерево, B⁺ дерево, 2-3-Tree, 2-3-4, RB, AA.
- (a) 25. Treap. Два определения случайного дерева (RBST). Эквивалентность определений. Оценка средней глубины вершины.
- (a) 26. Treap. $\exists!$ дерево при уникальных y . Связь со случайными деревьями. Операции Split, Merge.
- (b) 27. Treap. *Эффективная реализация Add, Delete.* Персистентные декартовы деревья.
- (b) 28. Treap. Garbage collector-ы на примере персистентного treap (ссылочный, стековый).

- (a) 29. Splay. Три вида поворотов. Операции Splay, Add. Формулировка теоремы о времени работы.
- (b) 30. Splay. Del, *Split*, *Merge*. Потенциал. Оценка всего, кроме операции Splay.
- (c) 31. Splay. Доказательство амортизированной оценки операции Splay.
- (c) 32. Splay. Оценка $\sum k_i \log \frac{m}{k_i}$ с доказательством.
- (c) 33. Splay. Теорема о статической оптимальности. Гипотеза динамической оптимальности.

Структуры данных

- (a) 34. Persistent. Offline, дерево версий.
- (a) 35. Persistent. *Детская персистентность (2 версии)*, *частичная персистентность* за $\mathcal{O}(\log n)$.
- (a) 36. Persistent. Online Fully Persistent массив за $\mathcal{O}(\log n)$. Любой способ.
- (b) 37. Persistent. Дек через Pairing.
- (b) 38. Persistent. *СНМ*. Персистентный массив: BST vs дерево отрезков.
- (c) 39. Persistent. Стек, очередь, дек за $\mathcal{O}(1)$ (через 5 стеков).
- (b) 40. Rope. Skip-List. Операции Find, Insert.
- (c) 41. Rope. Skip-List. Операции Erase, Split, Merge.
- (a) 42. SQRТ. Корневая декомпозиция. Статичная по массиву (с примером).
- (b) 43. SQRТ. Корневая декомпозиция. Через Split/Merge. Через Split/Rebuild. Примеры.
- (c) 44. SQRТ. Оптимальный выбор размера куска на примере задачи (`kth_stat + reverse`).
- (c) 45. *Корневая декомпозиция по запросам на примере «Dynamic Connectivity Offline за $\mathcal{O}(m\sqrt{m})$ »*.
- (a) 46. Дерево отрезков. Реализация сверху. Оценка на память, на время `get`.
- (a) 47. Дерево отрезков. Массовые модификации. `+=` и `getMin`. `=` и `getSum`.
- (b) 48. Дерево отрезков. Реализация снизу. Корректность. Сравнение с реализацией сверху.
- (b) 49. Дерево отрезков. Реализация сверху: ровно $2n$ ячеек.
- (b) 50. Дерево отрезков. Сравнение с BST по неявному ключу.
- (c) 51. Дерево отрезков. Динамическое (два способа), сжатие координат. Применение сжатия координат для задач со `scanline`.
- (a) 52. Scanline. 2D запрос на массиве через персистентную структуру для каждого префикса.
- (a) 53. Scanline. Число точек в прямоугольниках; число прямоугольников, накрывающих точку.
- (b) 54. Scanline. *Площадь объединения прямоугольников*.
- (b) 55. Scanline. *Наибольшая по сумме элементов возрастающая подпоследовательность*.
- (c) 56. Scanline. Лежит ли точка внутри невыпуклого многоугольника?
- (a) 57. 2D. ДО сортированных массивов. Связь с 2D запросом (кол-вом точек в прямоугольнике).
- (b) 58. 2D. Количество точек в прямоугольнике online за $\langle \mathcal{O}(n \log n), \mathcal{O}(\log n) \rangle$.
- (b) 59. 2D. Поиск k -й порядковой статистики на отрезке за $\mathcal{O}(\log^2 n)$.
- (c) 60. 2D. Многомерные структуры. ДО из ДО (зачем нужно?), ДО из ДД (зачем нужно?). Трёхмерное ДО, k -мерное ДО.
- (c) 61. 2D. Поиск k -й порядковой на отрезке за $\mathcal{O}(\log n)$.

RMQ, LCA, LA

- (a) 62. RMQ. Разреженные таблицы (Sparse Table).
- (b) 63. RMQ. Фенвик. Операции. Корректность.
- (a) 64. RMQ. Сведение LCA $\rightarrow$ RMQ ± 1 . Решение задачи LCA за $\langle \mathcal{O}(n \log n), \mathcal{O}(1) \rangle$.
- (b) 65. 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) 66. RMQ. *Решение задачи RMQ за $\langle \mathcal{O}(n \log^* n), \mathcal{O}(\log^* n) \rangle$* .
- (b) 67. RMQ. Сведение RMQ $\rightarrow$ LCA. Построение декартового дерева за линейное время.

- (c) 68. RMQ. Решение RMQ за $\langle \mathcal{O}(n), \mathcal{O}(1) \rangle$ для $n \leq w$, хранение стеков минимумов.
- (c) 69. RMQ. Фарах-Колтон-Бендер. Метод четырёх русских, решение RMQ и LCA за $\langle \mathcal{O}(n), \mathcal{O}(1) \rangle$.
- (c) 70. RMQ. *Фарах-Колтон-Бендер: предподсчёт ровно за $\mathcal{O}(2^k)$.*
- (a) 71. LCA. Двоичные подъёмы. Решение LCA, два способа (с 1-й двигающейся вершиной и с 2-мя).
- (b) 72. LCA в offline, алгоритм Тарьяна.
- (a) 73. LA. Решение LA в online и в *offline*.
- (b) 74. LA. Алгоритм Вишкина (решение LA за $\langle \mathcal{O}(n), \mathcal{O}(\log n) \rangle$).
- (a) 75. *Сумма на пути в дереве за $\langle \mathcal{O}(n), \mathcal{O}(1) \rangle$.*
- (c) 76. *Сумма на пути в меняющемся дереве за $\langle \mathcal{O}(n), \mathcal{O}(\log n) \rangle$.*

Другие древесные алгоритмы и структуры

- (a) 77. Heavy-Light Decomposition. Построение за $\mathcal{O}(n)$. Одно дерево отрезков.
- (a) 78. Heavy-Light Decomposition. Вычисление функции на пути, функции поддерева.
- (b) 79. Euler-Tour Tree. Операции Link, Cut, IsConnected.
- (b) 80. Link-Cut. Описание структуры. Операции Expose, MakeRoot, Link, Cut.
- (b) 81. Link-Cut. Потенциал. Оценка времени Expose.
- (c) 82. Link-Cut. Оценка времени MakeRoot, *Link*, *Cut*.
- (c) 83. MST за линейное время в среднем. Собственно алгоритм.
- (c) 84. MST за линейное время в среднем. Лемма про количество «небесполезных рёбер».

Игры

- (a) 85. Игры. Ациклический граф. Решение для симметричной/несимметричной игры.
- (b) 86. Игры. Граф с циклами. Ретроанализ. Реализация за $\mathcal{O}(E)$. Пример цикла без ничей.
- (c) 87. Игры. Длина самой короткой игры. Длина самой длинной игры.
- (b) 88. Игры. Ним. Результат игры Ним. Обоснование.
- (a) 89. Гранди. Функция Гранди, вычисление, связь с выигрышностью.
- (a) 90. Гранди. Прямая сумма. Теорема Гранди про ксор без доказательства.
Примеры на тему: Ним, *Игра в спички*, *Скамейки*.
- (b) 91. Гранди. Теорема Гранди про ксор: доказательство.
- (c) 92. Гранди. Максимальное значение функции Гранди.

Бонус

- (+) 93. Дополнительные деревья: ленивое полное дерево (перебалансировка при $2^k - 1$).
- (+) 94. Эппштейн. k -й по весу не обязательно простой путь за $(n + m + k) \log n$.
- (+) 95. Ориентированное MST. Алгоритм двух китайцев за $\mathcal{O}(m \log n)$.
- (+) 96. Персистентность через Fat Nodes. Решение задачи «2D-сумма на массиве» с предподсчётом за $\mathcal{O}(n)$ и запросом за $\mathcal{O}(\log n)$ в онлайн.
- (+) 97. Техника Fractional Cascading для двумерных запросов и параллельного бинарного поиска.
- (+) 98. КД-дерево. Ответы на двумерные запросы (и двумерные модификации!) за $\mathcal{O}(\sqrt{n})$, запрос «перечислить все точки в прямоугольнике за $\mathcal{O}(k + \log n)$ ».
- (+) 99. LA: Вишкин за $\langle \mathcal{O}(n \log n), \mathcal{O}(1) \rangle$, решение LA через ladders decomposition.
- (+) 100. Предподсчёт за $\mathcal{O}(n \cdot \alpha)$, позволяющий любой отрезок массива разбить на α непересекающихся предподсчитанных.
- (+) 101. LCA: двоичные подъёмы за $\mathcal{O}(n)$ памяти и быстрее по константе времени.
- (+) 102. Количество точек в полуплоскости, $\text{get}(a, b, c) = \text{count}(x : ax + by + c \leq 0)$
- (+) 103. HLD с запросами за $\mathcal{O}(\log n)$.