

**Вопросы к коллоквиуму по алгоритмам**  
**СПБ АУ, первый курс, весна, 2014/15 учебный год**

1. Дерево поиска: find, add, del за  $\mathcal{O}(h)$ . Хранение дерева. Обходы дерева: симметричный, прямой.
2. Дерево поиска: next, prev, find за  $\mathcal{O}(1)$ . Использование двусвязного списка, хеш-таблицы.
3. Случайное дерево поиска. Два эквивалентных определения. Оценка глубины.
4. AVL дерево: инвариант, оценка размера, глубины, малые и большие вращения, add, del.
5. AVL дерево: split за  $\mathcal{O}(\log^2 n)$ , merge за  $\mathcal{O}(\log n)$ .
6. Треар: единственность, глубина, split, merge, add, del за  $\mathcal{O}(\log n)$ ; build за  $\mathcal{O}(n \log n)$ .
7. Rope. Неявный ключ: идея, применение к AVL, treap. Sum, min/max, add, reverse на отрезке.
8. Skip-List (add, del, k-th, split, merge).
9. Persistent Tree, Persistent Array, ссылочный garbage collector, разбор задачи про “копирование памяти”
10. Persistent Stack, Queue за  $\mathcal{O}(1)$ .
11. RBST: split, merge, add, del за  $\mathcal{O}(\log n)$ ; build за  $\mathcal{O}(n)$ .
12. 2-3-tree, 2-3-4-tree, B-tree, RB-Tree, AA-Tree.
13. Статически оптимальное дерево поиска. Определение. Построение за  $\mathcal{O}(n^2)$ .
14. Дерево отрезков с операциями сверху, модификация в точке, на отрезке, динамическое дерево отрезков для диапазонов произвольной длины. Обработка запросов add(x), del(x), count(l, r) деревом отрезков.
15.  $k$ -я порядковая статистика, static data, online,  $\langle \mathcal{O}(n \log n), \mathcal{O}(\log n) \rangle$ .
16.  $k$ -я порядковая статистика, dynamic data, offline,  $\langle \mathcal{O}(n \log n), \mathcal{O}(\log^2 n) \rangle$  (и online за  $\mathcal{O}(\log n \cdot \log M)$ ).
17. 2D-orthogonal-query. Решение в online за  $\langle \mathcal{O}(n \log n), \mathcal{O}(\log^2 n) \rangle$  (дерево отрезков сортированных массивов, дерево merge-sort-a). Решение за  $\langle \mathcal{O}(n \log n), \mathcal{O}(\log n) \rangle$ .
18. Сведение задач двумерного поиска на массиве и двумерного поиска на плоскости друг к другу. Обработка  $d$ -мерного запроса деревом отрезков за  $\mathcal{O}(\log^d n)$ .
19. Дерево отрезков с операциями снизу. Дерево Фенвика. Многомерное дерево Фенвика.
20. Сканирующая прямая. 2D-orthogonal-query в offline за  $\langle \mathcal{O}(n), \mathcal{O}(n \log n) \rangle$ .  
Максимальная по весу возрастающая по обеим координатам последовательность точек за  $\mathcal{O}(n \log n)$ .
21. Сканирующая прямая. Площадь объединения прямоугольников.
22. Sparse Table, Disjoint Sparse Table, частичные суммы, полный предподсчёт. Классификация функций: какие функции какими из перечисленных структур считаются.
23. Модификации Sparse Table и Disjoint Sparse Table:  $\langle \mathcal{O}(n), \mathcal{O}(\log n) \rangle$ ,  $\langle \mathcal{O}(n \log \log n), \mathcal{O}(1) \rangle$ ,  $\langle \mathcal{O}(n), \mathcal{O}(\log \log n) \rangle$ ,  $\langle \mathcal{O}(n \log^* n), \mathcal{O}(\log^* n) \rangle$ .
24. Алгоритм Фараха-Колтона-Бендера: последовательность сведений и решение задачи RMQ $\neq$ 1 за  $\langle \mathcal{O}(n), \mathcal{O}(1) \rangle$
25.  $RMQ \rightarrow LCA$ . Построение Cartesian Tree за  $\mathcal{O}(n)$ .
26.  $LCA \rightarrow RMQ$ . Эйлеров обход дерева (как минимум 3 версии). Применение: функция на поддереве.
27. Алгоритм Тарьяна: LCA-Offline. Модификация алгоритма: RMQ-Offline.
28. Двоичные подъёмы, решение LCA за  $\langle \mathcal{O}(n \log n), \mathcal{O}(\log n) \rangle$ . Разреженная версия за  $\langle \mathcal{O}(n), \mathcal{O}(\log n) \rangle$ .  
Функции на путях не меняющегося дерева в online за  $\mathcal{O}(\log n)$ .
29. Функции на путях дерева в offline. Решение dfs-ом и деревом отрезков.  
Функции на путях дерева: биекция между вершинами и рёбрами.
30. Сумма на путях дерева в online. Случай неменяющегося дерева:  $\mathcal{O}(n), \mathcal{O}(1)$ .  
Случай меняющегося дерева:  $\mathcal{O}(n), \mathcal{O}(\log n)$ .
31. Минимум на путях дерева в offline за  $\mathcal{O}^*(1)$ . (!) *будет дорассказано 26-го марта.*
32. Рандомизированный алгоритм построения MST за  $\mathcal{O}(n + m)$ .
33. Euler-Tour-Trees (ET): link, cut, makeRoot, isConnected за  $\mathcal{O}(\log n)$ .
34. Heavy-Light-Decomposition (HLD): build за  $\mathcal{O}(n)$ , get за  $\mathcal{O}(\log^2 n)$ .
35. Задача LA. Решение в offline за  $\mathcal{O}(n + m)$ . Решения в online двоичными подъёмами, разреженными двоичными подъёмами, HLD. Идея: хранить можно только листья.
36. Задача LA. Longest-Path-Decomposition, Ladder-Decomposition, решение за  $\langle \mathcal{O}(n \log n), \mathcal{O}(1) \rangle$ . Микро-макро эвристика, решение за  $\langle \mathcal{O}(n), \mathcal{O}(1) \rangle$ .