

Вопросы к коллоквиуму по алгоритмам

СПБАУ, первый курс, весна, 2015/16 учебный год

Деревья поиска, деревья отрезков

1. (a) BST. add/del, find, next/prev, прошивка дерева. Add за $\mathcal{O}(h)$, все остальные операции за $\mathcal{O}(1)$.
2. (a) BST. Прямой, симметричный обходы; хранение дерева. Sort деревом, lowerbound на время add.
3. (b) BST. Дерево по неявному ключу, функция на отрезке, reverse на отрезке. Равные ключи в дереве.
4. (b) BST. AVL-Tree. Оценка высоты, добавление, удаление. Lm про одно вращение.
5. (b) BST. 2-3-Tree: добавление, удаление
6. (a) BST. Использование 2-3-Tree для получения (2-3-4, RB, B, AA)-Tree. Подробно про B-Tree.
7. (a) BST. Treap. RBST. Случайные деревья. Два способа делать add и del.
8. (a) BST. Treap: оценка средней глубины вершины и максимальной глубины вершины
9. (c) BST. Splay-Tree. Формулировки 3 теорем (и про бор!). Доказательство основной.
10. (b) Skip-List (add/del, split/merge). Интерфейс Rope.
11. (b) Персистентность: дерево, массив, СНМ. На примере AVL дерева.
12. (b) Персистентность в offline. Ссылочная сборка мусора для персистентных структур.
13. (a) Д.О. Реализации снизу, сверху: оценки времени, памяти, сравнение. Групповые операции.
14. (a) Д.О. Сжатие координат и динамическое (разреженное) дерево отрезков.
15. (b) Д.О. сортированных массивов. Д.О. of Д.О.
16. (a) ScanLine. Максимальная по весу 2D-возрастающая подпоследовательность.
17. (b) ScanLine. Обычный и персистентный. Обработка 2D-ортогональных запросов в offline и online.
18. (b) ScanLine. Площадь объединения прямоугольников за $\mathcal{O}(n \log n)$
19. (c) ScanLine. k -я статистика на отрезке в online за $\mathcal{O}(\log^2 n)$ без персистентности, за $\mathcal{O}(\log n)$.

Задачи RMQ и LCA

20. (a) RMQ. Случай меняющегося массива. Решение деревом отрезков. Lower bound.
21. (b) RMQ. Sparse Table. $\langle \mathcal{O}(n \log n), \mathcal{O}(1) \rangle$, $\langle \mathcal{O}(n), \mathcal{O}(\log \log n) \rangle$, $\langle \mathcal{O}(n \log \log n), \mathcal{O}(1) \rangle$, $\langle \mathcal{O}(n \log^* n), \mathcal{O}(\log^* n) \rangle$.
22. (a) RMQ \rightarrow LCA. Построение Cartesian Tree за $\mathcal{O}(n)$.
23. (a) LCA \rightarrow RMQ ± 1 . Эйлеров обход.
24. (b) RMQ. Фарах-Колтон-Бендер: последовательность сведений, решение RMQ ± 1 за $\langle \mathcal{O}(n), \mathcal{O}(1) \rangle$.
25. (b) Эйлеров обход: 3 версии и пример использования для каждой, функция на поддереве. IsAncestor.
26. (b) LCA. Offline алгоритм Тарьяна.
27. (b) RMQ. Модификация offline алгоритма Тарьяна для задачи RMQ.
28. (a) LCA. Двоичными подъёмами. Использование для вычисления функции на пути дерева.
29. (a) Функции на путях дерева. Offline решение dfs-ом и деревом отрезков.
30. (b) Функции на путях дерева. Online, сумма. Меняющееся и неменяющееся деревья.
31. (b) Функции на путях дерева. Минимум. 2 online решения. Offline: СНМ.
32. (a) LA. Offline решение. Алгоритм Вишкина за $\langle \mathcal{O}(n), \mathcal{O}(\log n) \rangle$.
33. (b) LA. Longest-path-decomposition, ladder-decomposition, алгоритм за $\langle \mathcal{O}(n \log n), \mathcal{O}(1) \rangle$.
34. (b) LA. Улучшение $\langle \mathcal{O}(n \log n), \mathcal{O}(1) \rangle$ до $\langle \mathcal{O}(n), \mathcal{O}(1) \rangle$.
35. (b) HLD. Heavy-Light-Decomposition.
36. (b) Centroid decomposition, функции на пути дерева.
37. (b) Euler-Tour-Trees. С переподвешиванием.
38. (c) Link-Cut-Tree. С доказательством для функции Expose.
39. (c) MST. Рандомизированный алгоритм за $\mathcal{O}(n + m)$. Без леммы.
40. (c) MST. Рандомизированный алгоритм за $\mathcal{O}(n + m)$. Лемма.

Паросочетания, раскраски

41. (a) Matching. Определения и сложность задач в двудольном и произвольном графе: паросочетание, контролирующее множество, вершинное покрытие, независимое множество, совершенное паросочетание. Матрица Татта (без док-ва).
42. (a) Matching. Лемма о дополняющем пути.
43. (a) Matching. Алгоритмы поиска паросочетания: обычный, Куна.
44. (b) Matching. Оптимизации алгоритма Куна:
не чистить пометки, вообще не чистить пометки, жадная инициализация, быстрое обнуление.
45. (b) Matching. Теорема Кёнига, поиск вершинного покрытия за $\mathcal{O}(V + E)$.
46. (b) Matching. Применение Куна для поиска паросочетания в произвольном графе.
47. (c) Matching. Классификация рёбер двудольного графа по принадлежности паросочетанию.
48. (b) Matching. Stable (marriage problem). Алгоритм за $\mathcal{O}(E)$.
49. (b) Покраски. Рёберная. Теорема Визинга. Связь с паросочетанием. Покраска рёбер двудольного графа.
50. (c) Покраски. Алгоритмы рёберной покраски двудольного графа за $\mathcal{O}(V^3 \log V)$ и $\mathcal{O}(E^2)$.
51. (b) Покраски. Вершинные. Брукс, теоремы о 5 и 4 красках, практически ценный алгоритм.

Правила сдачи

Вы записываетесь в [google-doc](#), получаете время сдачи, когда нужно прийти и сдать. В нужный момент вы приходите и говорите “я готов сдавать!”. Если есть свободный принимающий, в ответ вы получаете **три случайных числа** – номера вопросов в каждой из групп, и идёте **20 минут готовиться**. Если вы опоздали более чем на 5 минут, попадаете в живую очередь таких же опоздавших, у вас самый низкий приоритет, но как только кто-то осводится, вас тоже послушают.

При ответе билетов:

1. В **любых** спорных ситуациях подходим к лектору.
2. Лучше рассказать хоть что-то, чем ничего не сказать.
3. Вы готовились к экзамену! Вы должны хотя бы помнить, что вы должны знать, и что к какому билету относится.
4. Если вас спрашивают что-то не из билета, не из программы, не стесняйтесь говорить об этом.
5. Принимающий может в любой момент задать не сложный допвопрос из серии “что такое LCA?” или “за сколько вы умеете считать минимум на пути дерева?”.
6. Если вы плохо знаете свой билет, можно вкратце проговорить, что вы всё-таки по теме помните, и заменить билет. Замена происходит так: принимающий мгновенно генерит псевдослучайное число отличное от предыдущего и говорит вам. Замена билета – минуса к оценке: $(c) = -0.2$, $(b) = -0.4$, $(a) = -0.6$.
7. Заменять билет можно только один раз.
8. По каждой из тем у принимающего магическим образом складывается в голове оценка от 0 до 1 – уровень ваших знаний. Ваша оценка – сумма этих чисел, число от 0 до 3. Какие из этих чисел сообщать вам, а какие нет, решает принимающий. Сделанный вывод принимающий может описать не числом, а лишь на словах “тут вы почти ничего не знаете”, “а этот билет вы ответили идеально”.

Алгоритм выставления оценки:

- a) Меньше 1.33 – двойка. Ваши знания не удовлетворительны.
- b) Больше 2.66 – пятёрка. Вы круты.
- c) 2.2 и больше – точно четвёрка.
- d) 1.8 и меньше – точно тройка.
- e) Если от 1.8 до 2.2, то идёт серия простых и не очень допвопросов, помогающих уточнить оценку.