

Экзаменационные вопросы группы А

1. (3) Дерево отрезков: определение, построение. Реализация сверху запроса и обновления отдельного элемента.
2. (4) Дерево отрезков: модификация на отрезке.
3. (4) Дерево отрезков: решение задачи о K -ой порядковой статистике на отрезке за $O(\log^3 N)$.
4. (4) Двумерное дерево отрезков: реализация на матрице.
5. (4) Двумерное дерево отрезков: сжатое на N заранее известных точках.
6. (5) Двумерное дерево отрезков: прибавление на прямоугольнике.
7. (4) Дерево Фенвика: определение, построение. Многомерное дерево Фенвика. Формула включений-исключений.
8. (4) Дерево Фенвика: модификация на отрезке.
9. (4) Дерево Фенвика для необратимой операции.
10. (3) Декартово дерево: определение, построение. Построение из уже отсортированного (по ключам) массива за $O(n)$.
11. (4) Декартово дерево: операции Split, Merge, Find, Insert, Remove.
12. (4) Декартово дерево по неявному ключу: операции K -th, Split, Merge, Insert, Remove.
13. (4) Декартово дерево: групповые операции на отрезке.
14. (3) Skip-List. Операции Insert, Remove, Find.
15. (3) Система непересекающихся множеств: реализация лесом. Ранговая эвристика и эвристика сжатия путей. Доказательство оценки $O(n \log n)$ для ранговой эвристики.
16. (4) Система непересекающихся множеств: доказательство оценки $O(n \log^* n)$.
17. (4) Задача RMQ: Offline-алгоритм с использованием СНМ без сведения к задаче LCA.
18. (3) Задача LCA: постановка задачи. Наивный алгоритм. Метод двоичных подъемов ($n \log n, \log n$).
19. (4) Задача LCA: Offline-алгоритм Ахо-Хопкрофта-Ульмана-Тарьяна за $O(n \log^* n)$.
20. (4) Dynamic Connectivity Problem: Offline-алгоритм с использованием частично персистентной СНМ.
21. (4) Алгоритм Хьюи.
22. (4) Задачи LCA: сведение к задаче ± 1 RMQ за $O(n)$. Решение задачи RMQ методом Sparse Table ($n \log n, 1$).
23. (3) Метод четырёх русских на примере задачи о перемножении битовых матриц.
24. (5) Задача RMQ ± 1 : алгоритм Фараха-Колтона—Бендера ($n, 1$).
25. (4) Задача RMQ: сведение к задаче LCA за $O(n)$.
26. (3) Префикс-функция строки: алгоритм Кнута-Морриса-Пратта.
27. (4) Алгоритм Ахо-Корасик. Автомат Ахо-Корасик, использование.
28. (3) Z-функция.
29. (4) Простые строки. Алгоритм Дюваля.
30. (3) Хеширование строк. Использование хешей для сравнения строк.
31. (4) Суффиксный массив: определение, построение за $O(n \log n)$ (алгоритм Карпа-Миллера-Розенберга).
32. (3) Алгоритм Касаи-Ли-Аримур-Арикавы-Парка.
33. (3) Суффиксный массив: поиск образца в строке $O(m \log n)$.
34. (5) Суффиксный массив: построение за $O(n)$ (алгоритм Кяркяйнена-Сандерса).
35. (3) Суффиксные деревья: несжатые и сжатые, наивный алгоритм построения за $O(n^2)$.

36. (5) Суффиксные деревья: алгоритм Укконена построения суффиксного дерева за $O(n)$.
37. (3) Суффиксные деревья: применение в различных задачах.
38. (5) Суффиксный автомат: определение, построение за линейное время, доказательство оценки времени работы.
39. (4) Потоки и разрезы: определения, основные свойства, теорема и алгоритм Форда-Фалкерсона.
40. (4) Алгоритм Эдмондса-Карпа, доказательство оценки $O(VE^2)$.
41. (4) Алгоритм масштабирования потока, доказательство оценки $O(E^2 \log Max)$.
42. (4) Алгоритм Диница поиска блокирующего потока, реализация за $O(V^2E)$.
43. (3) Паросочетание в двудольном графе. Алгоритм Куна.
44. (4) Максимальный поток минимальной стоимости: теорема о потоке минимальной стоимости, использование алгоритма Форда-Белмана, его реализация в виде bfs.
45. (3) Игры на ациклических графах. Проигрышные и выигрышные позиции.
46. (3) Анализ игры с циклами, ретроанализ.
47. (4) Эквивалентность по Гранди, функция Гранди. Сумма игр.
48. (3) Проверка принадлежности точки произвольному многоугольнику за $O(n)$ с использованием луча и с использованием суммы углов за $O(n)$. Проверка принадлежности точки выпуклому многоугольнику за $O(\log n)$.
49. (4) Построение касательных к многоугольнику. Поиск расстояния от прямой до многоугольника.
50. (4) Вероятностный алгоритм нахождения точки в пересечении полуплоскостей за $O(n)$.
51. (4) Вероятностный алгоритм нахождения окружности минимального радиуса, содержащей заданные точки за $O(n)$.
52. (3) Алгоритм построения выпуклой оболочки. Алгоритм Грэхема.
53. (4) Алгоритм Мельмана построения выпуклой оболочки простой замкнутой ломаной.