

МФТИ, ФПМИ
Алгоритмы и структуры данных, 2-й семестр, весна 2023
Программа экзамена

Всюду, где уместно и не сказано иное, пункт программы подразумевает формулировку решаемой задачи, описание алгоритма, доказательство его корректности и анализ асимптотики.

1. Задача о кузнечике (набор максимальной суммы на массиве).
2. Задача о черепашке (набор максимальной суммы на таблице). Динамика «назад» и «вперёд».
3. Задача о наибольшей общей подпоследовательности. Восстановление ответа.
4. Задача о наибольшей возрастающей подпоследовательности: решение за $O(n \log n)$ с помощью бинарного поиска.
5. Задача о наибольшей возрастающей подпоследовательности: решение за $O(n^2)$.
6. Задача о наибольшей возрастающей подпоследовательности: решение за $O(n \log n)$ с помощью дерева отрезков или дерева Фенвика со сжатием координат.
7. Задача о рюкзаке: решения с динамикой по весу или по стоимости, что лучше выбрать. Восстановление ответа.
8. Задача о рюкзаке: сверхполиномиальность известных алгоритмов. Неоптимальность жадного алгоритма.
9. Бинарное возведение чисел и матриц в степень, асимптотика.
10. Подсчёт n -го числа Фибоначчи (по модулю m) за $O(\log n)$.
11. Подсчёт n -го члена рекуррентности $a_n = \lambda a_{n-1} + \mu a_{n-2} + 1$ (по модулю m) за $O(\log n)$ для произвольных констант λ и μ .
12. Количество путей длины ровно k между всеми парами вершинам за $O(n^3 \log k)$.
13. Количество путей длины не более k между всеми парами вершинам за $O(n^3 \log k)$.
14. Булево умножение матриц. Проверка наличия пути длины ровно k между всеми парами вершин за $O\left(\frac{n^3}{w} \log k\right)$.
15. Кодирование подмножеств $\{0, 1, \dots, n-1\}$ с помощью масок. Процедура извлечения бита (bit).
16. Операции над множествами (масками): объединение, пересечение, разность. Реализация в программе. Проверка, что одна маска является подмножеством другой.
17. Задача о максимальной клике: решения за $O(2^n \cdot n^2)$, $O(2^n \cdot n)$ и $O(2^n)$.
18. Задача о максимальной клике: решение за $O(2^{n/2})$.
19. Поиска хроматического числа графа за $O(2^n \cdot \text{poly}(n))$.
20. Поиск в глубину: алгоритм `dfs` на ориентированном графе. Лемма о белых путях.
21. Поиск в глубину: множество посещаемых вершин, поиск цикла, достижимого из s , проверка на ацикличность.
22. Топологическая сортировка ориентированного ациклического графа: определение и алгоритм поиска (с доказательством корректности). Число путей в ориентированном ациклическом графе.
23. Отношение сильной связности между вершинами. Компоненты сильной связности. Сильно связный граф.
24. Алгоритм Косарайю.
25. Постановка и решение задачи 2SAT (применение алгоритма выделения компонент сильной связности).
26. Определение эйлера цикла. Критерий наличия эйлера цикла/цикла в ориентированном/неориентированном графе (б/д).
27. Реализация алгоритма поиска эйлера цикла.
28. Алгоритм `dfs` на неориентированном графе. Дерево обхода `dfs`. Классификация рёбер на древесные и обратные.
29. Мосты, точки сочленения. Введение функции `ret`. Критерий того, что ребро является мостом. Критерий того, что вершина является точкой сочленения.

30. Насчёт **ret** в неориентированном графе, нахождение мостов и точек сочленения.
31. Определение расстояния в невзвешенном/взвешенном графе.
32. Поиск в ширину: алгоритм **bfs** с доказательством корректности.
33. Алгоритм **0-k-bfs**. Корректность — б/д.
34. Алгоритм Дейкстры. Условия применимости, доказательство корректности. Реализации за $O(n^2)$, $O(m \log n)$, $O(m + n \log n)$.
35. Алгоритм Форда—Беллмана: поиск кратчайших расстояний от одной вершины до всех. Реализация, асимптотика (в случае отсутствия отрицательных циклов).
36. Алгоритм Форда—Беллмана: нахождение кратчайших расстояний от одной вершины до всех в случае наличия отрицательных циклов.
37. Алгоритм Флойда: поиск попарных расстояний в графе без отрицательных циклов.
38. Восстановление ответа (пути) в алгоритме Флойда.
39. Алгоритм A^* : определения функций f, g, h . Реализация.
40. Допустимые и монотонные эвристики в алгоритме A^* . Примеры монотонных эвристик на разных сетях.
41. Формулировка работоспособности (корректность и время работы) алгоритма A^* в случае монотонной, допустимой или произвольной эвристики. Доказательство для монотонного случая.
42. Двусторонний **bfs**.
43. Остовный подграф, остовное дерево. Минимальный остов. Лемма о безопасном ребре.
44. Алгоритм Прима: доказательство корректности и реализации за $O(n^2)$, $O(m \log n)$, $O(m + n \log n)$.
45. Система непересекающихся множеств (СНМ). Виды запросов. Эвристика по рангу, эвристика сжатия путей. Асимптотика ответа на запрос при использовании обеих эвристик (б/д).
46. Асимптотика ответа на запрос в СНМ при использовании только эвристики по рангу.
47. Алгоритм Крускала: корректность, реализация, асимптотика.
48. Алгоритм Боруки: выбор минимального ребра из нескольких, корректность, реализация, асимптотика.
49. Определение паросочетания в произвольном графе, двудольного графа, увеличивающего пути.
50. Лемма об устройстве неориентированного графа, в котором степени всех вершин не превосходят двух.
51. Теорема Бержа.
52. Алгоритм Куна. Корректность, реализация, асимптотика.
53. Лемма об отсутствии увеличивающих путей из вершины при отсутствии таких путей относительно меньшего паросочетания.
54. Определения независимого множества, вершинного покрытия. Связь определений.
55. Алгоритм поиска максимального независимого множества и минимального вершинного покрытия в двудольном графе с помощью разбиения на доли L^-, L^+, R^-, R^+ (с доказательством). Теорема Кёнига.
56. Определения сети, потока, величины потока, остаточной сети. Пример, почему нельзя обойтись без обратных рёбер.
57. Определения разреза, величины разреза, величины потока через разрез. Лемма о равенстве величины потока и величины потока через разрез.
58. Теорема Форда—Фалкерсона.
59. Алгоритм Форда—Фалкерсона. Корректность, асимптотика. Пример сверхполиномиального (от размера входа) времени работы.
60. Алгоритм Эдмондса—Карпа. Корректность.
61. Лемма о возрастании $dist(s, v)$ между последовательными итерациями алгоритма Эдмондса—Карпа.
62. Лемма о числе насыщений ребра в алгоритме Эдмондса—Карпа. Асимптотика этого алгоритма.
63. Алгоритм Эдмондса—Карпа/Форда—Фалкерсона с масштабированием, асимптотика.
64. Определение слоистой сети, блокирующего потока. Алгоритм Диница, доказательство коррект-

ности.

65. Реализация алгоритма Диница. Асимптотика.
66. Первая теорема Карзанова о числе итераций алгоритма Диница.
67. Быстродействие алгоритма Диница в единичных сетях.
68. Алгоритм Хопкрофта—Карпа поиска максимального паросочетания в двудольном графе. Корректность и асимптотика.
69. Алгоритм Штор—Вагнера поиска минимального глобального разреза.
70. Min cost flow: постановка задачи. Критерий минимальности стоимости потока величины k . Алгоритм поиска потока величины k минимальной стоимости. Асимптотика.
71. Потенциалы Джонсона. Поиск min cost k -flow с помощью алгоритма Дейкстры.
72. Определение дерева, его свойства (б/д). Определение и поиск диаметра. Центр.
73. Определение центроида в дереве. Алгоритм поиска центроида в дереве. Лемма о количестве центроидов.
74. Определение изоморфизма графов. Алгоритм проверки изоморфности двух ориентированных или неориентированных деревьев за $O(n \log n)$.
75. Задача LCA. Постановка, решение с помощью двоичных подъёмов.
76. Задача LCA. Решение с помощью эйлерова обхода.
77. Задача LCA. Алгоритм Фарах-Колтона и Бендера.
78. Решение статической задачи RMQ (поиск минимума на отрезке массива) с предподсчётом за $O(n)$ и ответом на запрос за $O(1)$.
79. Центроидная декомпозиция. Поиск числа пар вершин в дереве на расстоянии d .
80. Heavy-light декомпозиция. Обновление значений и сумма на путях.
81. Link-cut tree: постановка задачи. Разбиение на пути, реализация **expose**. Асимптотика — б/д.