

МФТИ, ФИВТ
Алгоритмы и структуры данных, весна 2021
Программа экзамена

1. Пять шагов для решения задачи на динамическое программирование.
2. Задача о кузнечике (набор максимальной суммы на массиве). Неоптимальность жадного алгоритма.
3. Задача о черепашке (набор минимальной суммы на таблице). Динамика “назад” и “вперёд”.
4. Задача о наибольшей общей подпоследовательности.
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 наибольшей возрастающей подпоследовательности: решение за $O(n \log n)$ с помощью бинарного поиска.
8. Задача о рюкзаке: решения через $dp[i][w]$ и $dp[i][c]$, что лучше выбрать. Восстановление ответа.
9. Задача о рюкзаке: сверхполиномиальность известных алгоритмов. Неоптимальность жадного алгоритма.
10. Задача о рюкзаке: решение с $O(C)$ или $O(W)$ памяти без восстановления ответа.
11. Бинарное возведение чисел и матриц в степень, асимптотика.
12. Подсчёт n -го числа Фибоначчи (по модулю m) за $O(\log n)$.
13. 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 .
14. Количество путей между двумя вершинами длины ровно k за $O(n^3 \log k)$.
15. Нахождение $A + A^2 + \dots + A^k$ за $O(n^3 \log k)$ для матрицы A размера $n \times n$.
16. Количество путей между двумя вершинами длины не более k за $O(n^3 \log k)$.
17. Кодирование подмножеств $\{0, 1, \dots, n-1\}$ с помощью масок. Процедура извлечения бита.
18. Операции над множествами (масками): объединение, пересечение, разность. Реализация в программе. Проверка, что одна маска является подмножеством другой. Проверка, что число является степенью двойки.
19. Задача о самом дешёвом гамильтоновом пути: решение за $O(2^n \cdot n^2)$.
20. Задача о максимальной клике: решения за $O(2^n \cdot n^2)$, $O(2^n \cdot n)$ и $O(2^n)$.
21. Подсчёт всех значений $b(mask) = \max_{s \subset mask} a(s)$ для данного набор значений $a(0), \dots, a(2^n - 1)$ за $O(2^n \cdot n)$.
22. Задача о максимальной клике: решение за $O(2^{n/2} \cdot n)$.
23. Симпатичные узоры: количество раскрасок таблицы $n \times m$ в два цвета без одноцветных квадратов 2×2 . Прямой профиль: решения за $O(4^n \cdot (n + m))$ и $O(8^n \cdot \log m)$.
24. Симпатичные узоры. Изломанный профиль: идея решения за $O(2^n \cdot nm)$.
25. Определения неориентированного и ориентированного графов, пути, (вершинно) простого пути, рёберно простого пути. Связь вершинной и рёберной простоты. Определение цикла, рёберно простого цикла, (вершинно) простого цикла. Определение достижимости между вершинами, простота пути. Определение связности.
26. Три способа хранения графа в памяти компьютера, их преимущества и недостатки.
27. Поиск в глубину: алгоритм `dfs` на ориентированном графе. Лемма о белых путях.
28. Поиск в глубину: множество посещаемых вершин, поиск цикла, достижимого из s , проверка на ацикличность.
29. Топологическая сортировка ориентированного ациклического графа: определение и алгоритм поиска (с доказательством корректности).
30. Отношение сильной связности между вершинами. Компоненты сильной связности. Сильно связный граф.
31. Алгоритм Косарайю. Корректность и время работы.

32. Конденсация ориентированного графа, ацикличность.
33. Постановка и решение задачи 2SAT (применение алгоритма выделения компонент сильной связности).
34. Алгоритм `dfs` на неориентированном графе. Дерево обхода `dfs`. Классификация рёбер на древесные и обратные. Проверка связности и ацикличности. Компоненты связности.
35. Мосты, точки сочленения. Введение функции `ret`. Критерий того, что ребро является мостом.
36. Насчёт `ret` в неориентированном графе, нахождение мостов.
37. Нахождение точек сочленения в неориентированном графе.
38. Понятие рёберной двусвязности. Отношение эквивалентности.
39. Выделение компонент рёберной двусвязности в неориентированном графе. Древесность графа со сжатыми компонентами рёберной двусвязности.
40. Определение эйлерова цикла. Критерий наличия эйлерова цикла в неориентированном графе.
41. Реализация алгоритма поиска эйлерова цикла.
42. Критерий наличия эйлерова пути в неориентированном графе.
43. Критерий наличия эйлерова цикла в ориентированном графе.
44. Определение кратчайшего расстояния в невзвешенном/взвешенном графе.
45. Поиск в ширину: алгоритм `bfs` с доказательством корректности.
46. Алгоритм 0-K-`bfs`.
47. Двусторонний `bfs`.
48. Алгоритм Дейкстры. Условия применимости, доказательство корректности. Реализации за $O(n^2)$, $O(m \log n)$, $O(m + n \log n)$.
49. Двусторонний алгоритм Дейкстры. Завершение алгоритма: почему достаточно реализовать алгоритм, почему нельзя обойтись меньшим числом действий (пример).
50. Алгоритм A*: определения функций f, g, h ; реализация.
51. Вырожденные случаи в алгоритме A*: $h \equiv 0$, $h(v) = \text{dist}(v, t)$.
52. Допустимые и монотонные эвристики в алгоритме A*. Примеры монотонных эвристик на разных сетках.
53. Формулировка работоспособности (корректность и время работы) алгоритма A* в случае монотонной, допустимой или произвольной эвристики. Доказательство для монотонного случая.
54. Алгоритм Флойда: поиск попарных кратчайших расстояний в графе без отрицательных циклов. Реализация, асимптотика.
55. Восстановление ответа (пути) в алгоритме Флойда.
56. Алгоритм Форда—Беллмана: поиск кратчайших расстояний от одной вершины до всех. Реализация, асимптотика (в случае отсутствия отрицательных циклов).
57. Алгоритм Форда—Беллмана: нахождение кратчайших расстояний от одной вершины до всех в случае наличия отрицательных циклов.
58. Остовный подграф, остовное дерево. Минимальный остов. Лемма о безопасном ребре.
59. Алгоритм Прима: доказательство корректности и реализации за $O(n^2)$, $O(m \log n)$, $O(m + n \log n)$.
60. Система непересекающихся множеств (СНМ). Виды запросов. Эвристика по рангу, эвристика сжатия путей. Асимптотика ответа на запрос при использовании обеих эвристик (б/д).
61. Асимптотика ответа на запрос в СНМ при использовании только эвристики по рангу.
62. Алгоритм Крускала: корректность, реализация, асимптотика.
63. Алгоритм Борушки: выбор минимального ребра из нескольких, корректность, реализация, асимптотика.
64. Определение паросочетания в произвольном графе, двудольного графа, увеличивающего пути.
65. Лемма об устройстве неориентированного графа, в котором степени всех вершин не превосходят двух.
66. Теорема Бержа.
67. Алгоритм Куна. Корректность, реализация, асимптотика.
68. Лемма об отсутствии увеличивающих путей из вершины при отсутствии таких путей относительно

ного меньшего паросочетания.

69. Определения независимого множества, вершинного покрытия. Связь определений.
70. Алгоритм поиска максимального независимого множества и минимального вершинного покрытия в двудольном графе с помощью разбиения на доли L^-, L^+, R^-, R^+ (с доказательством).
71. Алгоритм поиска максимального независимого множества и минимального вершинного покрытия в двудольном графе с помощью задачи 2SAT.
72. Определения сети, потока, величины потока, остаточной сети. Пример, почему нельзя обойтись без обратных рёбер.
73. Определения разреза, величины разреза, величины потока через разрез. Лемма о равенстве величины потока и величины потока через разрез.
74. Лемма о связи величины произвольного потока и величины произвольного разреза.
75. Теорема Форда—Фалкерсона.
76. Алгоритм Форда—Фалкерсона. Корректность, асимптотика. Пример сверх-полиномиального (от размера входа) времени работы.
77. Алгоритм Эдмондса—Карпа. Корректность.
78. Лемма о возрастании $dist(s, v)$ между последовательными итерациями алгоритма Эдмондса—Карпа.
79. Лемма о числе насыщений ребра в алгоритме Эдмондса—Карпа. Асимптотика этого алгоритма.
80. Задача о разбиении коллектива на две группы с минимизацией суммарного недовольства.
81. Алгоритм Эдмондса—Карпа с масштабированием, асимптотика.
82. Определение слоистой сети, блокирующего потока. Алгоритм Диница, доказательство корректности.
83. Реализация алгоритма Диница. Асимптотика.
84. Первая теорема Карзанова о числе итераций алгоритма Диница.
85. Эффективность алгоритма Диница в единичных сетях.
86. Алгоритм Хопкрофта—Карпа поиска максимального паросочетания в двудольном графе. Корректность и асимптотика.
87. Алгоритм Штор—Вагнера поиска минимального глобального разреза.
88. Min cost flow: постановка задачи. Алгоритм поиска потока величины k минимальной стоимости (б/д). Асимптотика.
89. Критерий минимальности стоимости потока величины k .
90. Корректность алгоритма поиска min cost k -flow в отсутствие отрицательных циклов.
91. Потенциалы Джонсона. Поиск min cost k -flow с помощью алгоритма Дейкстры.
92. Задача о назначениях. Решение сведением к потоковой задаче.
93. Определение дерева, его свойства (б/д). Определение диаметра дерева. Алгоритм поиска диаметра в дереве.
94. Определение центроида в дереве. Алгоритм поиска центроида в дереве. Лемма о количестве центроидов (б/д).
95. Определение изоморфизма графов. Алгоритм проверки изоморфности двух ориентированных или неориентированных деревьев за $O(n \log n)$.
96. Задача LCA. Постановка, решение с помощью двоичных подъёмов.
97. Задача LCA. Решение с помощью эйлера обхода.
98. Задача LCA. Алгоритм Фарах-Колтона и Бендера.
99. Задача RMQ. Постановка, решение за $O(n)$ предподсчёта и $O(1)$ на запрос.
100. Heavy-light decomposition. Тяжёлые и лёгкие рёбра. Лемма о числе лёгких рёбер на пути между двумя вершинами. Решение задачи обновления на ребре и суммы на пути за $O(\log^2 n)$ на запрос.
101. Центроидная декомпозиция. Подсчёт числа объектов, обладающих заданным свойством.
102. Центроидная декомпозиция. Задача о перекрашивании синих вершин в красный цвет и поиска расстояния до ближайшей красной вершины.