

1. Структуры (C/C++)
2. Асимптотическая оценка сложности.
3. Определение стека.
4. Операции над стеком. Push и Pop.
5. Реализация на базе массива.
6. Задача про правильную скобочную последовательность, обратная польская нотация.
7. Определение очереди.
8. Операции над очередью. Реализация на базе массива. Зацикленный массив.
9. Дек и его реализация на базе массива.
10. vector. Операции resize, reserve, push_back
11. Постановка задачи сортировки. Порядок на элементах.
12. Сортировка вставками. Оценка сложности в лучшем и худшем случаях.
13. Сортировка выбором. Асимптотически точная оценка сложности.
14. Сортировка пузырьком. Оценка сложности в лучшем и худшем случаях.
15. Сортировка по сложному признаку. Хранение данных в структуре (записи) или в нескольких массивах. Функция сравнения элементов.
16. Сортировка подсчетом.
17. Постановка задачи поиска. Линейный поиск в неупорядоченном массиве.
18. Поиск самого левого и правого вхождений. Инвариант.
19. Бинпоиск по ответу. Пример задачи.
20. Бинарный поиск.
21. Вещественный бинпоиск.
22. Двоичная куча, определение, реализация на базе массива.
23. Бинарное дерево. Определения, задачи на всякого рода количества элементов и высоту дерева.
24. Основное свойство кучи. Просеивание элемента вверх, вниз.
25. Построение кучи. Доказательство или обоснование того, что это происходит за линейное время.
26. Изъятие минимального элемента.
27. Пирамидальная сортировка.

28. Добавление элемента. Просеивание элемента вверх.
29. Удаление произвольного элемента.
30. Быстрая сортировка. Общее описание подхода. Разбиение массива на две части.
31. Выбор барьерного элемента. Первый, средний. Случайный.
32. Оценка сложности
33. Лучший случай.
34. Худший случай.
35. Биномиальные коэффициенты. Определение, формула. Доказательство. Расчет по формуле.
36. Расчет при помощи рекурсии. Доказательство рекуррентного соотношения.
37. Общий подход к динамическому программированию.
38. Мячик. Количество путей.
39. Количество путей с препятствиями.
40. Путь максимальной/минимальной стоимости. С препятствиями.
41. Восстановление пути.
42. Черепашка. Количество маршрутов. $C(n + m - 2, n - 1)$.
43. Маршруты с препятствиями.
44. Путь максимальной стоимости. С препятствиями.
45. Вывод пути.
46. Наибольшая возрастающая подпоследовательность.
47. Определение подпоследовательности.
48. Решение за N^2 .
49. Наибольшая общая подпоследовательность.
50. Задача о рюкзаке.
51. Банкомат.
52. Найти количество способов замостить коридор размера $2 \times n$ плитками размера $1 \times K$, где $K \leq n$.
53. Дано сколь угодно много монет достоинством a_1, a_2, \dots, a_n рублей. Требуется определить, можно ли с их помощью набрать ровно s рублей. $O(ns)$.
54. Последовательности из 0 и 1 длины N .

55. Подсчет количества.
56. Номер по объекту.
57. Объект по номеру.
58. Генерация следующей последовательности.
59. Обобщение на случай K -ичной последовательности длины N .
60. Перестановки из N элементов.
61. Подсчет количества.
62. Номер по объекту.
63. Объект по номеру.
64. Генерация следующей перестановки.
65. Сочетания из N по K элементов.
66. Подсчет количества.
67. Генерация всех сочетаний.
68. Номер по объекту.
69. Объект по номеру.
70. Генерация следующего сочетания.
71. Граф, вершина, ребро, путь, простой путь, цикл, простой цикл, петля, кратные ребра, степень вершины.
72. Отношение достижимости. Компоненты связности.
73. Виды графов. Взвешенные, ориентированные. Полный граф. Двудольный граф.
74. Дерево, эквивалентные определения.
75. Сумма вершин с нечетными степенями.
76. Матрица смежности.
77. Списки ребер.
78. Списки смежных вершин.
79. Поиск в глубину.
80. Постановка задачи, стратегия обхода.
81. Метки цветов вершин (белые, серые, черные).
82. Рекурсивная реализация.

83. Классификация ребер дерева поиска в глубину.
84. Анализ времени работы для различных способов задания графа.
85. Поиск компонент связности.
86. Поиск циклов. Восстановление вершин цикла.
87. Раскраска графа в два цвета. Проверка на двудольность.
88. Топологическая сортировка.
89. Существование топологической сортировки.
90. С использованием поиска в глубину.
91. Поиск в ширину.
92. Кратчайшие пути в графе. Случай невзвешенного графа.
93. Стратегия обхода, описание алгоритма. Реализация.
94. Восстановление пути.
95. Анализ времени работы для разных способов представления графа.
96. Подсчет количества компонент связности.
97. Раскраска в два цвета, проверка на двудольность.
98. Поиск кратчайших путей от всех вершин до заданной.
99. Поиск кратчайших путей от нескольких источников. Введение фиктивной вершины.
100. Нахождение всех вершин/ребер, лежащих на кратчайшем пути между двумя вершинами. Два поиска в ширину навстречу.
101. Алгоритм Дейкстры.
102. Постановка задачи.
103. Релаксационная техника. Ребра отрицательного веса, циклы отрицательного веса.
104. Идея алгоритма, аналогия с поиском в ширину.
105. Реализация, доказательство корректности.
106. Вывод пути.
107. Оценка сложности для разных вариантов реализации.
108. Пример графа с отрицательными ребрами, на котором не работает Дейкстра.
109. Реализация при помощи кучи.
110. Алгоритм Флойда. Постановка задачи.

111. Рекуррентное соотношение.
112. Реализация. Обоснование уменьшения объема памяти до $O(N^2)$. Оценка сложности.
113. Нахождение циклов отрицательного веса.
114. Восстановление пути. Несколько способов.
115. Выигрышная и проигрышная позиции (определение, свойства)
116. Доказательство возможности восстановления четности в Ниме
117. Доказательство необходимости сменить четность с четной на нечетную в Ниме
118. Функция Гранди для одной игры (тех)
119. Сумма игр. Значение функции Гранди для суммы игр