

Яндекс Образование
Курс по олимпиадному программированию
Параллель В'

25.07.2023

Содержание

1	Сортировки	3
2	Бинарный поиск + тернарный поиск	3
3	Графы : знакомство	4
4	Динамическое программирование #1	4
5	C++	5
6	Структуры данных	5
7	Математика	5
8	BFS	6
9	DFS	6
10	Кратчайшие пути	6
11	Остовы	6
12	Динамическое программирование #2	7
13	Дерево отрезков #1	7
14	Коллоквиум	7
15	Комбинаторика #1	7
16	Корневая декомпозиция	7

17 Хеши	8
18 Строковые алгоритмы	8
19 Бор	8
20 Дерево отрезков #2	8
21 Сканирующая прямая	8
22 Практикум по графам	9
23 LCA	9
24 Динамическое программирование #3	9
25 Комбинаторика #2	9
26 Игры	10
27 Геометрия - примитивы	10
28 Геометрия - многоугольники	10
29 Паросочетания	11
30 Зачет	11

1 Сортировки

1. O-нотация. Математическое определение. Замечание о том, что для конкретных ограничений алгоритм, более быстрый асимптотически, не всегда действительно быстрее.
Сколько операций в секунду способен обрабатывать компьютер (порядка 10^8).
2. Навык по ограничениям в задаче понять, решения каких асимптотик найдут, а какие нет.
3. Квадратичные сортировки: пузырьком, выбором, вставками.
Сортировка подсчетом.
4. Quick и Merge sort
5. Определение стабильной сортировки. Какие из рассмотренных сортировок стабильны.

2 Бинарный поиск + тернарный поиск

1. Бинарный поиск. Понимание того, что для бинпоиска нужны:
 - а) Монотонная функция.
 - б) Инвариант для границ бинпоиска.
2. Бинарный поиск по ответу. Стандартные задачи с каким-нибудь процессом и просьбой минимизировать что-нибудь. Смежные с другими темами примеры: бинпоиск по ответу в графах.
3. Вещественный бинарный поиск. Замечание о том, что лучше всего устанавливать константное число итераций.
4. Утверждение о том, что непрерывная функция, принимающая в одной точке неположительное значение, а в другой – неотрицательное, имеет точку, в которой принимает нулевое значение. Применение бинпоиска для поиска этой точки.
5. Представление бинпоиска как поиска по степеням двойки: сначала пробуем установить самый старший бит в 1, и т.д.
Замечание о том, что можно делать такой поиск и в других системах счисления.
6. Galloring (экспоненциальный поиск) – поиск позиции элемента за $O(\log(i))$
7. Тернарный поиск. Примеры функций, к которым можно применять и к которым нельзя. Замечание о том, что, если функция не меняет своего значение лишь в точках экстремума, к ней все же можно применить тернарный поиск.

8. Бинарный поиск по производной.
9. Тернарный поиск с золотым сечением.

3 Графы : знакомство

1. Неориентированные графы.
Основные определения: вершина, ребро, путь, цикл, простой путь, простой цикл.
Изображение графов. Примеры.
Что можно представлять в виде графов? (Например, отношение дружбы.)
Ориентированные графы.
Взвешенные графы.
Как хранить графы?
 - a) Матрица смежности.
 - b) Списки смежности.
2. Взвешенные графы
3. Хранение графа списком ребер
4. Базовый DFS. Примеры:
 - a) Проверка графа на связность.
 - b) Выделение всех компонент связности.
5. Деревья. Определение дерева. Эквивалентные определения дерева.
Подвешенные деревья.
Как реализовывать DFS на деревьях без массива used?
6. DFS: время входа-выхода и топсорту.
7. Проверка на двудольность.
8. DFS: конденсация.

4 Динамическое программирование #1

1. Основы ДП. Кузнечик + черепашка (подсчет количества путей, подсчет минимального/максимального пути).
Формализуем сказанное: что нужно для ДП?
 - a) Какое состояние ДП? Что такое ответ для состояния (целевая функция)?
 - b) Какие начальные значения?
 - c) Как пересчитывать значения?
 - d) Какой должен быть порядок обхода, чтобы все значения, требуемые при пересчете, уже были подсчитаны?
 - e) Как получить ответ?

2. Восстановление ответа: через массив динамики и через массив предков.
3. Ленивая динамика: как удобно писать. Примеры задач, где без неё никак. Провести аналогии с DFS.
4. Динамика по состоянию последнего элемента: количество последовательностей из 0 и 1 без двух 0 подряд.

5 C++

6 Структуры данных

1. Куча: операции `sift_down`, `sift_up`. Как реализовывать `priority_queue`: извлечение минимума и добавление. Удаление/изменение веса произвольного элемента. Heap sort.
2. Стек, очередь, дек.
3. Вектор.
4. `set`, `map`.
5. `priority_queue`.
6. Префиксные суммы/минимумы/максимумы/...

7 Математика

1. Деление с остатком.
2. Сравнение по модулю и его свойства.
3. НОД и НОК.
4. Алгоритм Евклида.
5. Бинарный алгоритм Евклида.
6. Расширенный алгоритм Евклида.
7. Обратный по модулю. Критерий существования.
8. Применение расширенного алгоритма Евклида для поиска обратного по произвольному модулю.
9. Нахождение всех делителей числа за $O(\sqrt{n})$.
10. Факторизация за $O(\sqrt{n})$.

11. Быстрое возведение в степень.
12. Малая теорема Ферма.
13. Применение малой теоремы Ферма к поиску обратного по простому модулю.
14. Решето Эратосфена за $O(n \log \log n)$.
15. Решето Эратосфена за $O(n)$.
16. Применение решета Эратосфена для факторизации чисел за $O(\log n)$ с предсчетом за $O(n)$.
17. Определение вероятности. Подсчет простых вероятностей.
18. Матожидание. Его линейность.

8 BFS

1. BFS.
2. 0-k BFS (в частности, 0-1 BFS = deque).

9 DFS

1. 2-SAT.
2. Мосты и точки сочленения.
3. Эйлеровы пути и циклы.

10 Кратчайшие пути

1. Дейкстра: базовая и на сети (priority_queue!).
2. Форд-Беллман.
3. Флойд.
4. Детекция циклов отрицательного веса.

11 Остовы

1. Минимальные остовы.
2. Алгоритм Прима.
3. Алгоритм Крускала.
4. СММ.

12 Динамическое программирование #2

1. Рюкзак. Взвешенный/невзвешенный. Число предметов ограничено/не ограничено.
Реализация на двумерном массиве. Реализация на одномерном массиве (храним лишь текущий слой): замечание о том, что решения для ограниченного и неограниченного количества предметов отличаются направлением цикла for.
2. Динамика по префиксам. Примеры:
 - а) НВП.
 - б) Количество строк заданной длины из какого-то ограниченного алфавита, не содержащих другую заданную строку в качестве подпоследовательности.
3. Динамика по последнему взятому элементу: НВП.

13 Дерево отрезков #1

1. Дерево отрезков с операцией изменения в точке.

14 Коллоквиум

15 Комбинаторика #1

1. Правила суммы и произведения.
2. Перестановки. Факториал как количество перестановок длины n .
3. Циклы в перестановках. Четность перестановки через циклы, через инверсии. Обратная перестановка. Композиция перестановок.
4. Треугольник Паскаля.
5. Бином Ньютона.

16 Корневая декомпозиция

1. Корневая декомпозиция массива для нахождения функции на отрезке.
2. Корневая в задачах на строки: разбиение строк на короткие и длинные.
3. Корневая на графах: тяжелые и легкие вершины.
4. Корневая декомпозиция по запросам.

17 Хеши

1. Хеши.
2. Парадокс дней рождения.
3. Доказательство парадокса дня рождений для 23 человек

18 Строковые алгоритмы

1. Префикс-функция.
2. Z-функция.
3. Алгоритм Манакера

19 Бор

1. Бор. Способы хранения: map, unordered_map (один большой), массив, вектор.
2. Цифровой бор

20 Дерево отрезков #2

1. Дерево отрезков с групповыми операциями.
2. Дерево отрезков: поиск подотрезка с максимальной суммой.
3. Дерево отрезков: применения
НВП, Дейкстра, Хеши, ДП

21 Сканирующая прямая

1. Сканлайн. Просто.
Задача о нахождении длины объединения отрезков.
Задача о выборе максимального количества непересекающихся отрезков.
2. Сканлайн. Средне.
Задача о нахождении суммы в прямоугольнике.
Задача о поиске точки, покрытой максимальным количеством прямоугольников.
Задача о поиске площади объединения прямоугольников.
Задача о поиске количества различных на отрезке.
3. Сжатие координат

22 Практикум по графам

1. Работа с неявно построенными графами. Преимущество по затратам памяти
2. Примеры задач: роботы в лабиринтах, отношения и т.д."

23 LCA

1. Sparse Table.
2. Двоичные подъемы:
 1. Аналогия с бинарным поиском.
 2. Задача LA.
 3. Задача LCA.

24 Динамическое программирование #3

1. Динамика по подстрокам/подотрезкам. Примеры:
 - а) Для каждой подстроки строки узнать, является ли она палиндромом; найти количество подпалиндромов; найти максимальный подпалиндром.
 - б) Для выпуклого многоугольника найти триангуляцию минимальной суммарной длины.
2. Динамика по подмножествам. Подробности в "Биты, ДП по маскам"
3. Динамика по цифрам. Примеры:
 - а) Количество чисел от 1 до n , в которых цифры неубывают.
 - б) Количество чисел от 1 до n , кратных k , в которых цифры неубывают.

25 Комбинаторика #2

1. Поиск числа сложных комбинаторных объектов через динамику: разные виды ПСП, разные разбиения на слагаемые, разные последовательности со странными свойствами (убывающие, возрастающие).
2. Генерация всех комбинаторных объектов в лексикографическом порядке: перестановок, сочетаний, размещений (с повторениями и без), ПСП, убывающие-возрастающие последовательности, разбиения на слагаемые (разные виды).
3. Поиск объекта по номеру. Поиск номера по объекту.

26 Игры

1. Выигрышные, проигрышные состояния.
2. Теория Шпрага-Гранди.
3. Ретро-анализ.

27 Геометрия - примитивы

1. Структура вектор.
2. Скалярное и векторное (псевдоскалярное) произведения.
3. Площадь многоугольника.
4. Расстояние от точки до прямой.
5. Проекция точки на прямую.
6. Общее уравнение прямой. Переход от него к двум точкам и наоборот. Вектор нормали.
7. Пересечение прямых.
8. Пересечение прямой и окружности.
9. Пересечение двух окружностей.
10. Поиск касательных к окружности.

28 Геометрия - многоугольники

1. Проверка на принадлежность точки многоугольнику за $O(n)$. Два способа: сумма углов и луч.
2. Алгоритм Джарвиса.
3. Алгоритм Грэхема.
4. Алгоритм Эндрю.
5. Локализация точки в выпуклом многоугольнике за $O(\log n)$ на запрос.
6. Поиск касательных из точки к выпуклому многоугольнику за $O(\log n)$ на запрос.
7. Пересечение прямой с выпуклым многоугольником за $O(\log n)$ на запрос.

29 Паросочетания

1. Паросочетания. Алгоритм Куна.
2. Лемма Холла
3. Минимальное вершинное покрытие и максимальное независимое множество в двудольных графах.

30 Зачет