1. Бинпоиск (15 минут)

1.1) Постановка простейшего вида задач - дан отсортированный массив элементов, найти в нем данный или определить, что такого нет

1.2) Алгоритм бинпоиска

1.3) Оценка асимптотики

1.4) Обобщение постановки - дана функция $f(x)$, хотим найти точку $x0$, такую, что $f(x0) = y0$

1.5) Пример задачи: даны коэффициенты уравнения $ax^3+bx^2+cx+d = 0$, известно, что у него ровно 1 решение. Найти это решение. Делаем бинпоиск по $x0$, если знак многочлена в $x0$ совпадает со знаком a, то сдвигаем правую границу, иначе левую. Формально: $f(x) = ax^3 + bx^2 + cx + d$, ищем точку, где $f(x) = 0$.

2. Бинпоиск по ответу (25 минут)

2.1) Общая постановка задач - имеется монотонная функция f, ищем самую левую/самую правую точку $x0$ такую, что $f(x0) \leqslant (\geqslant) y0$

2.2) Алгоритм бинпоиска по ответу

2.3) Первый пример: даны n точек с целыми координатами на прямой, выбрать из них k таких, что минимальное расстояние между соседними как можно больше. Сначала посортируем точки. Будем бинпоиском искать это оптимальное минимальное расстояние. Пусть зафиксировано расстояние d, тогда можем за один проход по точкам проверить, можем ли взять хотя бы k так, чтобы между соседними расстояние было $\geqslant d$. Если смогли, то сдвигаем нижнюю границу, иначе верхнюю. Формально, работаем с функцией $f(d)$, которая равна максимальному количеству взятых точек, если расстояние между соседними не может быть меньше d. Тогда мы ищем такое максимальное d, что $f(d) \geqslant k$

*2.4) Второй пример: дан неориентированный граф, для каждого ребра определены две величины: t - время проезда по нему, и w - максимальный вес, который выдерживает это ребро. Нужно за время T успеть провести как можно больший вес из вершины 1 в вершину N, причем использовать можно только ребра, которые выдерживают выбранный вес. Бинпоиском ищем оптимальный вес, пусть он фиксирован и равен w. Проверим, что такой вес можем провести. Для этого оставим в графе только ребра, которые выдерживают w, и в нем найдем кратчайшее время проезда от вершины 1 в вершину N с помощью Дейкстры. Если оно $\leqslant T$, то сдвигаем нижнюю границу, иначе верхнюю. Формально: функция $f(w)$ - за какое минимальное время можем доехать из вершины 1 в N, если везем груз весом w. Ищем максимальное w такое, что $f(w) \leqslant T$.

3. Особенности бинпоиска в вещественных числах (5 минут)

3.1) Смена условия на фиксированное количество итераций

3.2) Оценка необходимого количества итераций

4. Тернарный поиск (35 минут)

4.1) Постановка задачи

4.2) Описание алгоритма

4.3) Доказательство корректности

4.4) Оценка асимптотики

4.5) Первый пример: есть n велосипедистов, для i-го задано начальное положение s_i и скорость v_i (все движутся вправо). Найти такой момент времени t, что разница между самым правым и самым левым велосипедистами на текущий момент как можно меньше. Делаем тернарник по t - времени, $f(t)$ - расстояние между самым правым и самым левым, эту функцию мы минимизируем.

4.6) Тернарник на плоскости

4.7) Второй пример: задан треугольник на плоскости, найти точку, минимизирующую сумму расстояний до его вершин. Введем функцию $f(x, y)$ - сумму расстояний от точки (x, y) до вершин треугольника, мы ее минимизируем. Введем $g(x0)$ - минимум $f(x, y)$ для всех точек, для которых $x = x0$. Тогда задача решается двумя вложенными тернарниками: внешний по x, минимизирующий $g(x)$, внутренний по $f(x, y)$, где x уже фиксирован.

*5. Золотое сечение (5 минут)

5.1) Определение

5.2) Применение в тернарном поиске - дает возможность не вычислять одну из точек деления на всех шагах, кроме первого.

*6. Двоичные подъемы (15 минут)

6.1) Постановка задачи: для списка нужно быстро находить i-ый элемент справа от j-ого, элементы добавляются/удаляются слева.

6.2) Идея двоичных подъемов

6.3) Применение к поставленной задаче

6.4) Оценка асимптотики

**6.5) Использование двоичных подъемов для поиска LCA