

1. *Бинарный поиск*

Корень из n . Полиномиальность языка квадратов натуральных чисел.

2. *Динамическое программирование*

Задача о рюкзаке.

Классическая постановка: Рюкзак 0-1 (0-1 Knapsack Problem). Задача формулируется как задача оптимизации. Имеется n грузов. Для каждого i -го груза определён его вес $w_i > 0$ и ценность $c_i > 0$. Ограничение суммарного веса предметов в рюкзаке задаётся грузоподъёмностью W . Необходимо максимизировать $\sum_i c_i x_i$ при ограничениях $\sum_i w_i x_i \leq W$, $x_i \in \{0, 1\}$.

Так же можно сформулировать задачу распознавания языка KNAPSACK, которая заключается в проверке того, что из подмножества из n элементов для каждого из которых задан вес w_j и стоимость c_j , мы можем взять такое подмножество $B \subset \{1, \dots, n\}$, что $\sum_{i \in B} w_i \leq W$ и $\sum_{i \in B} c_i \geq C$. Она в каком-то смысле не сложнее, чем вышеприведенная задача оптимизации.

Рассмотрим другой вариант постановки: Неограниченный целочисленный рюкзак. Каждого предмета можно брать сколько угодно и известно, что $w_i \in \mathbb{N}$. Тогда задачу можно решить с помощью динамического программирования за время $O(nW)$. Пусть $C[w]$ - максимальная суммарная ценность предметов, которые можно поместить в рюкзак грузоподъёмностью w . Тогда $C[0] = 0$ и $C[w] = \max_i (c_i + C[w - w_i])$. Но сложность не полиномиальна по размеру входа! Если W очень большое, то вход имеет размер $\log W + \dots$, а в асимптотику входит W .

3. *NP и начало про полиномиальную сводимость*