Московский физико-технический институт
Физтех-школа прикладной математики и информатики

Б О Л Ь Ш О Е Н А З В А Н И Е К У Р С А

V СЕМЕСТР

Лектор: *Иван Иванович Иванов*

h\nu

Автор: *Павел Дуров*
Проект на Github

осень 2022

Содержание

1 Динамическое программирование

<u>Задача:</u> Пусть есть полоска $1 \times n$, где в $i-$ой клетке записано число a_i. В нулевой клетке находится кузнечик, способный прыгать на 1 или 2 позиции вправо. Хотим найти максимальную сумму

Доказательство.
Достаточно заметить, что попав на i-ую позицию, предыдущие прыжки никак не повлияют на максимальное значение с начальной точкой в текущей позиции.
Решение:

1. Заведем массив dp, где в dp[i] будет хранится максимальная сумма до данной клетки (то есть из всевозможных путей выбираем наибольший)

2. Запишем в $\text{dp}[0] = 0$, в $\text{dp}[1] = a_1$

3. Пусть k клеток заполнены. Тогда $k+1$-ая будет пересчитываться по формуле

$$\text{dp}[k+1] = a_{k+1} + max(\text{dp}[k], \text{dp}[k-1])$$

4. Тогда наш ответ равен значению $\text{dp}[n]$

   -Асимптотика O(n)

□

Доказательства во всех задачах ДП проводятся по индукции по шагу алгоритма
Общая концепция:
Придумываем, что хранить → Пишем пересчет → Находим ответ в конце

<u>Задача с ЕГЭ:</u> Есть таблица $n \times m$, где в каждой клетке написана ее цена. Хотим найти максимальный путь из нижнего левого угла в правый верхний.
P.S. двигаемся только вверх или вправо
Решение:

1. Создаем массив $n \times m$, где в каждой клетке хранится наибольшая цена среди путей до этой клетки.

2. Записываем во всех "крайних клетках"сумму на единственном пути до нее.

3. Для остальных клеток формула пересчета такая:

$$dp\left[i\right]\left[j\right] = a_{ij} + max(dp\left[i-1\right]\left[j\right], dp\left[i\right]\left[j-1\right])$$

4. Получаем ответ в клетке $dp[n][m]$

   -Асимптотика O(nm)

<u>**Еще задачка:**</u> НОП (наибольшая общая последовательность)
Ищем наибольшую по длине общую последовательность в двух s и t.

1. Пусть $\text{dp}[i]\left[j\right]$ - длина НОП для последовательностей $s_{1,2,\ldots,i}$ и $t_{1,2,\ldots,j}$

2. dp[0] [$\circ$] = 0, dp[$\circ$] [0] = 0

3. Хотим найти dp[i] [j]:

   (a) s_i не участвует в НОП $\to$ dp[$i-1$] [j]

   (b) t_j не учатствует в НОП $\to$ dp[i] [$j-1$]

   (c) s_i и t_j участвуют в НОП, тогда они должны быть равны и ответ: dp[$i-1$] [$j-1$] + 1

<u>И еще одна:</u> НВП (наибольшая возрастающая последовательность)
Решение 1:

1. dp[i] [k] - минимальное значение элемента, на котором заканчивается последовательности длины k, если рассматривать только элементы $a_1, a_2, \ldots, a_i$

2. dp[0] [0] $= -\infty, dp[0][k > 0] = +\infty$

3. Пусть известна dp[$i-1$] [$\circ$]

   Далее, есть 2 случая:

   (a) Не берем a_i, тогда ответ равен dp[$i-1$] [j]

   (b) Берем a_i Тогда найдем $min\ k$, что dp[i] [k] $\geqslant a_i$ и поменяем значение на a_i

   Заметим, что выполняется инвариант:

   $$\text{dp}[i]\,[0] < \text{dp}[i]\,[1] < \ldots < \text{dp}[i]\,[k]$$

   А тогда, кроме dp[i] [k] под условие б) ничего не подойдет, а еще, это k можно найти с помощью бин поиска.

Таким образом, ДП в этой задаче будет заполняться построчно, где $i+1$-ая строка получается из i-ой изменением одного элемента
Асимптотика $O(n \log n)$
Решение 2:
Оживляем элементы по возрастанию, предварительно стабильно отсортировав их. dp[i] - максимальная длина ВП, оканчивающейся в a_i на момент оживления этого элемента.

a_i	2	3	1	5	4	6	5
$dp\,[i]$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$

$\to$

a_i	2	3	1	5	4	6	5
$dp\,[i]$	$\times$	$\times$	1	$\times$	$\times$	$\times$	$\times$

a_i	2	3	1	5	4	6	5
$dp\,[i]$	1	$\times$	1	$\times$	$\times$	$\times$	$\times$

$\to$

a_i	2	3	1	5	4	6	5
$dp\,[i]$	1	2	1	$\times$	$\times$	$\times$	$\times$

a_i	2	3	1	5	4	6	5
$dp\,[i]$	1	2	1	$\times$	3	$\times$	$\times$

$\to$

a_i	2	3	1	5	4	6	5
$dp\,[i]$	1	2	1	3	3	$\times$	4

a_i	2	3	1	5	4	6	5
$dp\,[i]$	1	2	1	3	3	4	4

Несложно заметить, что $dp\,[i]$ будет равно максимальному значению слева от текущей ячейки, что мы умеем находить за $O(\log n)$ через ДО

Тогда итог (максимальное значение в таблице) будет найдено за $O(n \log n)$

Последняя: (Рюкзак)

Есть n предметов, w_i - вес i-го элемента, а c_i - его стоимость. Вместимость рюкзака - W. Найти максимальную стоимость содержимого.

Обозначим за $dp\left[i\right]\left[a\right]$ максимальную стоимость предметов, если выбирать какие-то предметы из первых i с суммой веса a.

Тогда аналогично с предыдущей задачей будем вычислять $dp\left[i+1\right]\left[a\right]$, выбирая a_{i+1} или не выбирая его.

Это будет соответствовать значениям $dp\left[i\right]\left[a-w_i\right]+c_i$ и $dp\left[i\right]\left[a\right]$

Тогда поскольку $a \in 0, 1, \ldots, W$, окончательная асимптотика будет равна $O(nW)$