

# Динамическое программирование

Алексей Владыкин

СПбГУ ИТМО

25 марта 2011

- Ричард Беллман, 1950-е гг.
- В данном контексте «программирование» значит «метод оптимизации»: см. целочисленное программирование, линейное программирование, квадратичное программирование. . .
- Принцип оптимальности
- Уравнение Беллмана

$$V(x) = \max_{a \in \Gamma(x)} \{F(x, a) + \beta V(T(x, a))\}$$

- Необходимые свойства задачи:
  - оптимальная подструктура задачи (optimal substructure)
  - перекрывающиеся подзадачи (overlapping subproblems)
- Типичные классы задач:
  - оптимизация некоторой целевой функции
  - подсчет количества решений

## Наибольшая общая подпоследовательность

- Даны конечные последовательности  $s$  и  $t$ ;  $|s| = m$ ;  $|t| = n$
- Найти последовательность  $x$  наибольшей длины, являющуюся подпоследовательностью  $s$  и  $t$
- Пусть  $f(i, j)$  — длина наибольшей общей подпоследовательности  $s_1, \dots, s_i$  и  $t_1, \dots, t_j$ . Тогда

$$f(i, j) = \begin{cases} 0, & \text{если } i = 0 \text{ или } j = 0, \\ f(i - 1, j - 1) + 1, & \text{если } s_i = t_j, \\ \max(f(i - 1, j), f(i, j - 1)), & \text{если } s_i \neq t_j. \end{cases}$$

- Длина искомой наибольшей подпоследовательности —  $f(m, n)$ . Сама подпоследовательность находится обратным проходом по таблице значений  $f$ .

## Редакционное расстояние (расстояние Левенштейна)

- Даны строки  $s$  и  $t$ ;  $|s| = m$ ;  $|t| = n$
- Найти кратчайшую последовательность операций вставки, удаления и замены одного символа, преобразующую  $s$  в  $t$
- Пусть  $d(i, j)$  — редакционное расстояние между  $s_1, \dots, s_i$  и  $t_1, \dots, t_j$ . Тогда

$$d(i, j) = \begin{cases} 0, & \text{если } i = j = 0, \\ i, & \text{если } i \neq 0, j = 0, \\ j, & \text{если } i = 0, j \neq 0, \\ \min(d(i, j - 1) + 1, \\ d(i - 1, j) + 1, \\ d(i - 1, j - 1) + (s_i \neq t_j)). \end{cases}$$

- Искомое расстояние —  $d(m, n)$ .

## Целочисленная задача о рюкзаке

- Дан набор  $n$  вещей с весами  $w_1, \dots, w_n$  и стоимостями  $v_1, \dots, v_n$ . Дан рюкзак, вмещающий максимальный вес  $w$ .
- Найти подмножество вещей максимальной стоимости, которое можно унести в рюкзаке.
- Пусть  $v(i, j)$  — максимальная стоимость набора вещей, в который входят только вещи до  $i$ -й, и вес которого не превышает  $j$ . Тогда

$$v(i, j) = \begin{cases} 0, & \text{если } i = 0 \text{ или } j = 0, \\ v(i-1, j), & \text{если } j < w_i, \\ \max(v(i-1, j), & \text{если } w_i \leq w \\ v(i-1, j-w_i) + v_i). & \end{cases}$$

- Стоимость искомого набора —  $v(n, w)$ .

## Перемножение набора матриц

- Даны матрицы  $A_1, \dots, A_n$  размеров  $(p_1, q_1), \dots, (p_n, q_n)$ .
- Найти последовательность умножений матриц в выражении  $A_1 \cdots A_n$ , при котором количество умножений чисел минимально.
- Пусть  $m(i, j)$  — минимальное число умножений чисел, необходимое для вычисления  $A_i \cdots A_j$ . Тогда

$$m(i, j) = \begin{cases} 0, & \text{если } i = j, \\ \min_{i \leq k < j} (m(i, k) + m(k + 1, j) + p_i q_k q_j). \end{cases}$$

- Количество умножений в искомой последовательности —  $m(1, n)$ .

# Триангуляция выпуклого многоугольника

- Дан выпуклый  $n$ -угольник с вершинами  $v_1, \dots, v_n$  и весовая функция  $w$ .
- Найти такую триангуляцию данного  $n$ -угольника, что сумма  $w$  по всем треугольникам минимальна.
- Пусть  $m(i, j)$  — минимальный вес триангуляции многоугольника, построенного на вершинах  $i, \dots, j$ . Тогда

$$m(i, j) = \begin{cases} 0, & \text{если } j - i \leq 1, \\ \min_{i < k < j} (m(i, k) + m(k, j) + w(\Delta v_i v_k v_j)). & \end{cases}$$

- Вес оптимальной триангуляции —  $m(1, n)$ .

## Замощение площадки костяшками домино

- Дана площадка размера  $m \times n$  клеток и неограниченное число костяшек домино размера  $1 \times 2$ .
- Найти количество способов полного замощения площадки костяшками.
- Пусть  $c(i, p)$  — количество способов полностью замостить столбцы  $0 \dots i - 1$  и замостить столбец  $i$  с *профилем*  $p$ . Тогда

$$c(i, p) = \begin{cases} 1, & \text{если } i = 0 \text{ и } p = 0, \\ 0, & \text{если } i = 0 \text{ и } p \neq 0, \\ \sum_{p'} c(i - 1, p') d(p', p), & \end{cases}$$

где  $d(p', p)$  — количество способов получения профиля  $p$  из профиля  $p'$  в соседних столбцах.

- Искомое количество замощений —  $c(n, 0)$ .

## Рекомендуемая литература

-  Ахо А., Хопкрофт Дж., Ульман Дж.  
Структуры данных и алгоритмы. : Пер. с англ. : Уч. пос.  
М.: Издательский дом «Вильямс», 2000. — 384 с.: ил. // Глава 10
-  Кормен Т., Лейзерсон Ч., Ривест Р.  
Алгоритмы: построение и анализ.  
М.: МЦНМО, 1999. — 960 с., 263 ил. // Глава 16
-  Столяр С. Е., Владыкин А. А.  
Информатика: Представление данных и алгоритмы.  
СПб.: Невский Диалект; М.: БИНОМ. Лаборатория знаний,  
2007. — 382 с.: ил. // Глава E