Динамическое программирование

Иван Казменко

Кружок обучения мастерству программирования при мат-мехе СПбГУ

Четверг, 2 мая 2013 года

Оглавление

Новая модель телефона: постановка задачи

- Есть n этажей и k телефонов ($1 \leqslant n \leqslant 100\,000$, $0 \leqslant k \leqslant n$).
- Зафиксировано целое число x ($1 \leqslant x \leqslant n$). Его требуется найти.
- Одно действие — бросить телефон с этажа y ($1 \leqslant y \leqslant n$).
- Если $y \geqslant x$, телефон сломался, его нельзя больше использовать.
- Если $y < x$, телефон не сломался, его можно бросать снова.
- За какое минимальное количество ходов можно найти x?

Новая модель телефона: решение

- Пусть $f(n, k)$ — минимальное количество ходов, за которое можно различить n этажей, использовав не более k телефонов.
- База: $f(n, 1) = n - 1$, $f(1, k) = 0$.
- Заметим, что если $k > \log n$ (двоичный логарифм), то этаж x можно искать просто двоичным поиском. Поэтому интересных состояний — порядка $n \log n$.
- Переход: выполним одно действие.

Новая модель телефона: решение (продолжение)

- Переход: выполним одно действие.
- Мы выбираем этаж y наилучшим образом.
- Учтём, что окружающая среда может выбрать один из двух вариантов — сломался телефон или нет — наихудшим для нашей стратегии образом.
- Если телефон сломался, $x \leqslant y$, и нам осталось различить y нижних этажей, используя $k - 1$ телефон.
- Если телефон не сломался, $x > y$, и нам осталось различить $n - y$ верхних этажей, используя те же k телефонов.
- Формула: $f(n, k) = 1 + \min\limits_{y=1..n-1} \max\{f(y, k-1), f(n-y, k)\}$.

Новая модель телефона: код

```
for n = 2..N:
  for k = 1..K:
    f[n][k] = ∞
    for y = 1..n - 1:
      cur = 1 + max (f[y][k - 1], f[n - y][k])
      f[n][k] = min (f[n][k], cur)
```

- Это решение работает за $O(n^2 k)$.
- Учтём, что решать задачу имеет смысл, только если $k > \log n$, иначе ответ — это $\lceil \log n \rceil$.
- После разбора этого случая решение будет работать за $O(n^2 \log n)$.

Новая модель телефона: оптимизация

- Можно ускорить это решение до $O(nk \log n)$.
- Если учесть оценку $k \leqslant \log n$, получается $O(n \log^2 n)$.
- Рассмотрим части внутри выражения как функции от y.
- $f(y, k-1)$ — нестрого возрастающая функция от y.
- $f(n-y, k)$ — нестрого убывающая функция от y.
- $\max\{f(y, k-1), f(n-y, k)\}$ — сначала нестрого убывающая, а затем нестрого возрастающая функция от y.
- Значит, точку, где у этой функции достигается минимум, можно каждый раз искать, не перебирая все значетия y за $O(n)$, а двоичным поиском за $O(n \log n)$.

Новая модель телефона: оптимизация (продолжение)

- Можно ещё ускорить решение до $O(nk)$.
- Если учесть оценку $k \leqslant \log n$, получается $O(n \log n)$.
- Посмотрим, что общего в вычислении $f(n, k)$ и $f(n+1, k)$.
- $s(y) = \max\{f(y, k-1), f(n-y, k)\}$ — сначала нестрого убывающая, а затем нестрого возрастающая функция от y.
- $t(y) = \max\{f(y, k-1), f(n+1-y, k)\}$ — почти такая же функция, она в каждой точке не меньше, чем $s(y)$.
- Значит, точка минимума функции $t(y)$ не меньше, чем точка минимума функции $s(y)$.
- А значит, можно вычислять $f(n, k)$ для фиксированного k и сразу для всех n в сумме за $O(n)$, применив метод двух указателей: значение y с ростом n меняется только в бóльшую сторону.

Новая модель телефона: оптимизированный код

```
s (y, n, k):
  return max (f[y][k - 1], f[n - y][k])

for k = 1..K:
  y = 1
  for n = 2..N:
    while (y + 1 < n) AND (s (y, n, k) > s (y + 1, n, k)):
      y += 1
    f[n][k] = 1 + s (y, n, k)
```

To be continued...