SPb HSE, ПАДИИ, 1 курс, весна 2023/24
Практика по алгоритмам #7

Динамика-2

29 февраля

Собрано 1 марта 2024 г. в 10:56

Содержание

Динамика-2

1. **Магазины**

   На прямой есть n городов, в i-м живёт a_i жителей, находится город в точке x_i.
   Выбрать k городов, открыть в них магазины, чтобы минимизировать суммарное по всем жителям «расстояние до ближайшего магазина».

2. **Динамика по подотрезкам: свёртка**

   Дана строка из латинских букв длины n, нужно ее запаковать в максимально короткую, используя правило $n(S) = \underbrace{SS\ldots S}_{n}$.

   Пример: `NEERCYESYESYESNEERCYESYESYES` $\to$ `2(NEERC3(YES))`.

3. **Разбиения на слагаемые**

   Сколько способов разбить число n на k неупорядоченных различных слагаемых? Т.е. сколько есть возрастающих массивов длины k с суммой ровно n.

4. **Задача Иосифа Флавия**

   Древняя игра. n человек стоят по кругу. Считалочка. Каждый p-й по счету человек покидает круг и сбрасывется в яму с крокодилами. Даны n и p, кто останется в кругу последним? $\mathcal{O}(n)$.
   Пример $n = 6, p = 2$. Люди уходят в порядке $2, 4, 6, 3, 1$.

5. **Игра с камнями**

   Есть кучка из n камней. Двое играют в игру. Кто не может ходить, проиграл. Кто выиграет при оптимальной игре обоих? За ход можно брать любое число камней из $A = \{a_1, a_2, \ldots a_k\}$.

6. **Динамика по дереву**

   a) Дано дерево. Насчитайте размеры всех поддеревьев.
   b) Найти максимальное по весу паросочетание (веса на рёбрах) за $\mathcal{O}(n)$.

7. **Пожмножества**

   a) Как кодировать множество битмасками?
   b) Гамильтоно путь.
   c) Найдите количество топсортов за $2^n \cdot m$.
   d) Для каждого множества вершин в графе проверьте, независимо ли оно за 2^n.
   e) `(*)` Найти самую большую клику за 2^n, за $2^{n/2}$.

8. `(*)` **Доминошки**

   Дано поле 16×16. Сколько способов замостить его доминошками 1×2?

Разбор задач практики

1. **Магазины**

   См. конспект ПМИ.

   По сути мы разбиваем n точек на k отрезков. i-й отрезок – кто, в i-й магазин пойдёт. $f[i, j]$ – стоимость разбить первые i точек на j отрезков, в каждом отрезке установить 1 магазин, отправить людей с отрезка в него. $f[i, j] = \min_{L<i}(f[L, j-1] + w[L, i]$ – перебрали левую границу последнего отрезка $[L, i)$, здесь $w[L, i)$ – предподсчёт, вес отрезка, если туда поставить оптимальный магазин.

2. **Динамика по подотрезкам: свёртка**

   Главная идея – динамика по подотрезкам. Как закодировать отрезок $[i, i+len)$? Переберём все варианты, как закодировать первый символ. Отдельно? Тогда откусим его, закодируем остаток $[i+1, i+len)$. В виде «строка $[i, i+m)$, повторённая k раз»? Если так, откусим первые mk символов, закодируем их отдельно, остаток отдельно... если остаток не пуст, это корректный переход в динамике, а если остаток пуст, значит $mk = len$ и отрезок $[i, i+len)$ имеет период m. Описанное выше выражается через два перехода: $\forall m$ попробовать откусить первых m символов или разделить весь отрезок на кусочки длины m.

   Собственно решение за $\mathcal{O}(n^3)$.
   1. Предподсчитаем `lcp[i,j]` за $\mathcal{O}(n^2)$.
   2. Динамика `ans[i,len]` – ответ на задачу для подстроки `[i..i+len)`.
   `ans[i,len]` мы минимизируем ⇒ используем функцию `relax(&a,b) { a = min(a,b); }`.
   Пары $\langle i, len\rangle$ перебираем в порядке возрастания len.
   Для каждой пары изначально `ans[i,len] = len` (не сжимаем подотрезок).

   Далее в цикле по m делаем переходы двух типов:
   - «откусить первые m символов» и
   - «повторить подстроку длины m $\frac{len}{m}$ раз».

   Проверка, что, повторив $[i, i+m)$, получим $[i, i+len)$, делается так: `lcp[i,i+m]` $\geqslant$ `len-m`.

3. **Разбиения на слагаемые**

   Простая динамика.
   Рекурсивный перебор, перебираем слагаемые в порядке возрастания `rec(n,k,x)` – уже набрали сумму n из k слагаемых, последнее взятое x, тогда `rec(n,k,x)` $= \sum_{y>x}$ `rec(n+y, k+1, y)`.
   Добавили запоминание, получили динамику. Ускорение: `rec(n,k,x)` = `rec(n+x,k+1,x+1)` + `rec(n,k,x+1)` (или берём, или не берём x, как в рюкзаке, мы – рюкзак на весах $1, 2, \ldots$).

   Быстрая динамика.
   Либо слагаемых меньше k, либо из каждого из них можно вычесть единичку.
   $p[n, k] = p[n, k-1] + p[n-k, k]$.

4. **Задача Иосифа Флавия**

   Когда из круга уйдёт один человек, останется круг из $n-1$ человека. В этом круге последним уйдёт человек номер `f[n-1]`. В исходном круге он имел номер `f[n] = (f[n-1]+p)%n`.

5. **Игра с камнями**

   $f[i]$ – есть ли гарантированный способ выиграть у того, кто ходит, когда в кучке i камней.
   $f[i] = !f[i-a_1] \vee !f[i-a_2] \ \vee \ \ldots \ \vee !f[i-a_k]$.

6. **Динамика по дереву**

   a) $size[v] = 1 + \sum_x size[x]$, где x — дети v.

   b) На дереве за $\mathcal{O}(n)$: считаем две величины
   $f_0[v]$ – вес максимального паросочетания в поддереве v, не покрывающего v.
   $f_1[v]$ – вес максимального паросочетания в поддереве v, покрывающего v.

   Если нам неважно, покрывать вершину v или нет, то будем писать $f[v] = \max(f_0[v], f_1[v])$.
   Пусть дети вершины v – $x_1, x_2, \ldots, x_k$, веса рёбер (v, x_i) – w_i, тогда:
   $f_0[v] = \sum_i f[x_i]$
   $f_1[v] = \sum_i f[x_i] +$ ребро вниз $= \sum_i f[x_i] + \max_i \big(w_i - f[x_i] + f_0[x_i]\big)$

7. **Пожмножества**

   a) Как кодировать множество битмасками? См. конспект ПМИ и видеозапись лекции.
   b) Гамильтоно путь. См. конспект ПМИ и видеозапись лекции.
   c) Найдите количество топсортов за $2^n \cdot m$. $f[A] = \sum_{v:\, g[v] \subseteq A = \varnothing} f[A \,|\, 2^v]$. Где $g[v]$ – рёбра, входящие в v.
   d) Для каждого множества вершин в графе проверьте, независимо ли оно за 2^n. См. конспект ПМИ.
   e) (*) Найти самую большую клику за 2^n, за $2^{n/2}$.
   См. конспект ПМИ.

8. (*) **Доминошки**

   См. конспект ПМИ.

Домашнее задание

1. **(2+1) Сколько существует счастливых билетов?**

   Счастливый билет – строка из $2n$ цифр такая, что сумма цифр в первой половине равна сумме цифр во второй. Нужно решение за $\mathcal{O}(n^2)$.

   (a) Сколько существует чисел длины n с фиксированной суммой S?
   (b) Сколько существует счастливых билетов?

3.1. Дополнительная часть

1. **(2) Различные и далёкие**

   Дан массив объектов. У каждого объекта есть стоимость $cost_i$ и тип $type_i \leqslant k$.

   Выберите множество объектов максимальной суммарной стоимости так, что у каждой пары объектов типы различны, а разность индексов хотя бы d. $n \leqslant 100$, $k \leqslant 10$.