

蟻本P.319～

# 分割統治法

# 目次

- 分割統治法とは
- 列の分割統治法
- 木の分割統治法
- 平面の分割統治法
- 計算量解析・その他

# 分割統治法とは

そのままでは解決できない大きな問題を小さな問題に分割し、その全てを解決することで、最終的に最初の問題全体を解決する、という問題解決の手法である。（Wikipedia）

## 要はDP

# 列の分割統治法

P.319

## バブルソートの交換回数

長さ  $n$  の順列が与えられます

バブルソートでソートするのに必要なスワップ回数を求めてください

$$1 \leq n \leq 10^5$$

## 競プロ典型：転倒数

ご存じBinary Indexed Tree を利用することで  $\Theta(n \log n)$  で求めることができます

```
for i,e in enumerate(a):  
    sum += i - BIT.sum(0, e)  
    ++BIT[e]
```

# 分割統治による方法

与えられた順列  $A$  を  $B, C$  に二等分する  
各  $x < y$  は,

|                        |              |
|------------------------|--------------|
| $x \in B$ かつ $y \in C$ | 転倒していない      |
| $x \in C$ かつ $y \in B$ | がんばる         |
| $x, y \in B$           | $B$ について同じ問題 |
| $x, y \in C$           | $C$ について同じ問題 |

のどれか

下二つの再帰の深さは、毎回長さが半分になるので  $\Theta(\log n)$  回

$B, C$  がソート済みであるとする、前から見ていってマージしながら数え上げることができる

よって、各深さでは  $\Theta(n)$  で数え上げられる

したがって、マージソートしながら数えれば  $\Theta(n \log n)$

# 木の分割統治法

木に対しても同じように分割統治法を適用できないかを考える

木は適当な場所で切ってもバラバラになったそれぞれが木のままなので, いい感じのところで切って同じように再帰的な処理をすることができそう



# 重心分解

木の重心：その頂点を取り除いたとき、残った最大部分木の頂点数が最小になるようなもの

よって、木の重心を取り除くと残った部分木それぞれの頂点数は元々の木の  $\frac{1}{2}$  以下になる

よって、重心で分けて列と同じように分割統治することで再帰の深さが  $O(\log n)$  になる

実装がだるい

## 重心の探し方

適当な頂点  $v$  を木の根とする

$v$  の子  $u$  について,  $u$  を根とする部分木の頂点数が  $\frac{n}{2}$  より大きいようなものが高々一つ存在する

存在しなければ  $v$  が木の重心

存在すれば,  $u$  について同じことをする

P.321

## Tree (POJ 1741)

$n$  頂点の木が与えられる

最短距離が  $k$  以下である頂点の組の数を求めよ

$$1 \leq n \leq 10^4$$

実装

サンプル(1個)しか試してないですC++98ってなに

重心分解には大きく二種類

- 木全体での数え上げ
- 木上のクエリ

があり, 実装方針がかなり異なる

木全体での数え上げ : 重心ごとに毎回dfsをする

木上のクエリ : セグ木のような感じで重心から重心へ伝播させる

## 「みんなのプロコン 2018」 決勝

### C - 木の問題

$N$  頂点の木が与えられる 次のクエリ  $Q$  個に答えよ

- 頂点  $v$  からの距離が  $k$  である頂点の個数を答えよ

$$1 \leq N, Q \leq 10^5$$

$$0 \leq k \leq N - 1$$

クエリを一気に処理する

各ステップについて, 重心を  $c$  とする

$c$  からdfsをして, 距離ごとの頂点数を求める

各クエリについて  $c$  を通るような条件をみたすパスの数を数える  
(通らないパスは次の再帰で求める)

$$\Theta((N + Q) \log N)$$

実装

## Codeforces Round #199 (Div. 2)

### E. Xenia and Tree

$n$  頂点の木が与えられ, 頂点 1 は赤, 残りは青で塗られている  
次の二種類のクエリ  $m$  個を処理せよ

- 頂点  $v$  を赤く塗る
- 頂点  $v$  から最も近い赤く塗られた頂点までの距離を答える

$$2 \leq n \leq 10^5$$

$$1 \leq m \leq 10^5$$

重心からその重心を取り除いてできた部分木の重心への辺を張ったグラフを考えると, 高さ  $O(\log N)$  の木ができる

$d_v =$  頂点  $v$  から最も近いかもしれない赤く塗られた頂点までの距離

クエリ 1 : 頂点  $v$  から構築した木を上りながら  $d_v$  を更新する

クエリ 2 : 頂点  $v$  から構築した木を上りながら最短距離を計算する

$\Theta(N \log N + Q(\log N)^2)$  (LCAの実装によっては  $\log$  が落ちる)

実装



# 平面の分割統治法

P.324

## 最近点对問題

平面上の  $n$  個の点について, 最も近い二点の距離を求めよ

$$1 \leq n \leq 10^4$$

なんか乱択で線形とかもあるらしい  
(ソースらしきブログが見れなくなってたのでよくわからん)

同じように半分にして再帰的に解くことを考える

$x$  座標の中央値で平面を左右に分割する

二点  $p, q$  について

どちらも同じ領域に含まれているなら再帰的に計算する

違う領域に含まれていたら？

再帰的に計算したほうの最小距離を  $d$  と置く

これを更新し得るものだけ考えればいい

異なる領域に含まれ, かつ距離が  $d$  未満になり得るものは？

$x$  座標  $y$  座標別々に考えてみると

## $x$ 座標

分割した場所からの距離が  $d$  未満のものだけを考えればよい  
よって,  $x$  座標の中央値を  $x_m$  として,  $x$  座標が  
 $x_m - d < x < x_m + d$  である点のみを考えればよい

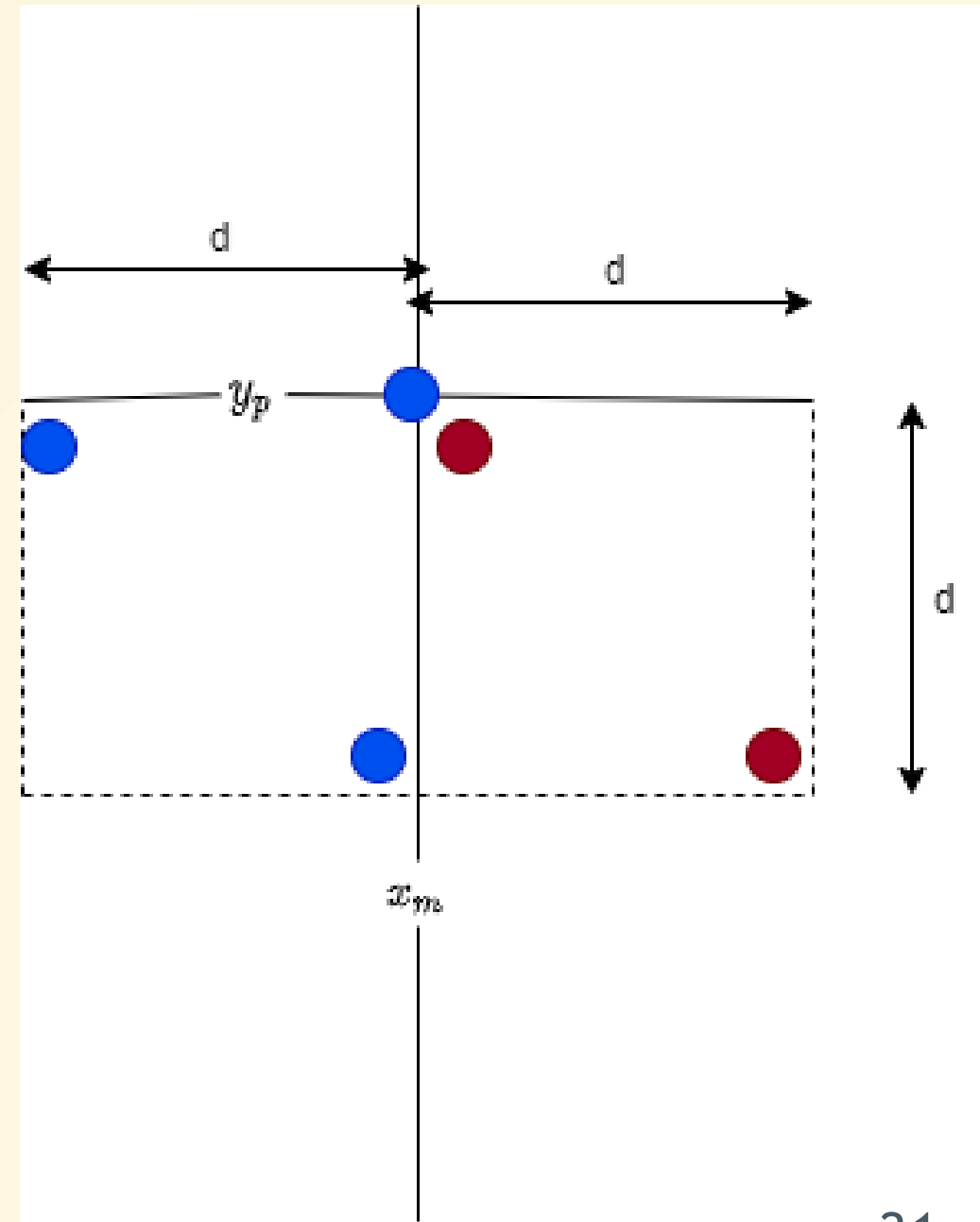
## $y$ 座標

各点について,  $y$  座標が自分以下かつ距離が  $d$  未満のものだけ考える  
としてよい  
よって, 見ている点の  $y$  座標を  $y_p$  として  
 $y$  座標が  $y_p - d < y \leq y_p$  である点のみを考えればよい

$2d \times d$  の長方形領域に含まれる点の数は高々 6 個である

よって、各点について定数時間で処理できる

$y$  座標でソートされていると嬉しいので、 $x$  座標の中央値で分割して  $y$  座標の順にソートするマージソートを行いながら計算することで合計  $\Theta(n \log n)$  となる



蛇足：

## 数学的な計算量解析

分割・統合が線形時間の場合の計算量

$$a \geq 1, b > 1, c, d > 0$$

$$T(n) = \begin{cases} aT\left(\frac{n}{b}\right) + cn \\ d \quad (n \leq 1) \end{cases} \Rightarrow T(n) = \begin{cases} \Theta(n^{\log_b a}) & (a > b) \\ \Theta(n \log n) & (a = b) \\ \Theta(n) & (a < b) \end{cases}$$

## 証明

$$\begin{aligned}T(n) &= aT\left(\frac{n}{b}\right) + cn \\&= a\left(aT\left(\frac{n}{b^2}\right) + c\frac{n}{b}\right) + cn \\&= \dots \\&= a\left(\dots a\left(aT\left(\frac{n}{b^k}\right) + c\frac{n}{b^{k-1}}\right) \dots + c\frac{n}{b}\right) + cn \\&= a^k d + cn\left(\left(\frac{a}{b}\right)^{k-1} + \dots + \frac{a}{b} + 1\right)\end{aligned}$$



$a > b$  とすると

$$\begin{aligned} &= a^k d + cn \frac{\left(\frac{a}{b}\right)^k - 1}{\frac{a}{b} - 1} \\ &< a^k d + \frac{cn}{\frac{a}{b} - 1} \left(\frac{a}{b}\right)^k \\ &= Ca^k \\ &= Ca^{\log_b n} \\ &= Cn^{\log_b a} \end{aligned}$$

カラツバ法

$n$  桁の整数の乗算  $X \times Y$  を計算する

$X = aR + b, Y = cR + d$  とすると

$$XY = acR^2 + (ac + bd - (a - b)(c - d))R + bd$$

と書ける よって

$$T(n) = 3T\left(\frac{n}{2}\right) + \Theta(n)$$

となり  $T(n) = \Theta(n^{\log_2 3})$

## 凸法

蟻本に載っているのはくグラハムスキャンで  $\Theta(n \log n)$

分割統治でも解ける

半分に分けてそれぞれで凸法, 上と下の接線で結合 を繰り返す

$$T(n) = 2T\left(\frac{n}{2}\right) + \Theta(n)$$

となりこれも  $\Theta(n \log n)$

## SA-IS

線形の大雑把ソートで半分くらいそろえた後

残った大きさ半分以下の同じ問題を再帰的に解く

$$T(n) = T\left(\frac{n}{2}\right) + \Theta(n)$$

となり  $\Theta(n)$

縮小法とよんだりもする？ recursive slowdown とかいう記述も

## 計算量

分割・統合が定数時間

$$a \geq 1, b > 1, c, d > 0$$

$$T(n) = \begin{cases} aT\left(\frac{n}{b}\right) + c \\ d \quad (n \leq 1) \end{cases} \Rightarrow T(n) = \begin{cases} \Theta(n^{\log_b a}) & (a > 1) \\ \Theta(\log n) & (a = 1) \end{cases}$$

$n = b^k$  とすると簡単に証明できます

木分解

<https://qiita.com/drken/items/4b4c3f1824339b090202>

最近点对

<http://www.cs.ucsb.edu/~suri/cs235/ClosestPair.pdf>

凸法

<https://www.jaist.ac.jp/~uehara/course/2016/i431/pdf/04greedy.pdf>

問題集

<https://blog.hamayanhamayan.com/entry/2017/05/21/234616>