# 完全单调性

**in $\leq 60$ minutes**

郭晓旭

http://icpc-camp.org

ICPCCamp 2016 Day 7

# Roadmap

1. 基础定义
2. $O(n \log n)$ Row Minima
3. 应用
4. $O(n)$ Row Minima (SMAWK)
5. Online Row Minima

# Monge Property

$$n \times m \text{ 的矩阵 } M_{n\times m}$$

$$\forall a \le b, \forall c \le d$$

$$M(a,c) + M(b,d) \le M(a,d) + M(b,c)$$

# Geometric Explanation

# Test

$$\forall a \leq b, \forall c \leq d \; M(a,c) + M(b,d) \leq M(a,d) + M(b,c)$$
$$\iff M(i,j) + M(i+1,j+1) \leq M(i,j+1) + M(i+1,j)$$

# Proof

# Total Monotonicity

$$\forall a \leq b, \forall c \leq d$$

$$M(a, c) \geq M(a, d) \implies M(b, c) \geq M(b, d)$$

# MP $\implies$ TM

如果 $M(a,c) \geq M(a,d)$ and $M(b,d) \geq M(b,c)$

$M(a,c) + M(b,d) \geq M(a,d) + M(b,c)$ 和 MP 矛盾

所以 $M(b,c) \geq M(b,d)$

# Row Minima

令 $A(i) = \arg\min_j M(i,j)$

则 $\forall a \leq b\ A(a) \leq A(b)$

# Proof

$$\forall c \leq A(a) \; M(a,c) \geq M(a, A(a))$$

$$\forall b \geq a \; M(b,c) \geq M(b, A(a))$$

$$\implies A(b) \geq A(a)$$

# Proof

$$\forall c \leq A(a)\ M(a, c) \geq M(a, A(a))$$

$$\forall b \geq a\ M(b, c) \geq M(b, A(a))$$

$$\implies A(b) \geq A(a)$$

# $O(m \log n)$ for RM

Find $A(\lfloor \frac{n}{2} \rfloor)$ in $O(m)$

$A(i) \leq A(\lfloor \frac{n}{2} \rfloor)$ for $1 \leq i \leq \frac{n}{2}$

$A(\lfloor \frac{n}{2} \rfloor) \leq A(j)$ for $\frac{n}{2} \leq j \leq n$

# $O(m \log n)$ for RM

Find $A(\lfloor \frac{n}{2} \rfloor)$ in $O(m)$

$A(i) \leq A(\lfloor \frac{n}{2} \rfloor)$ for $1 \leq i \leq \frac{n}{2}$

$A(\lfloor \frac{n}{2} \rfloor) \leq A(j)$ for $\frac{n}{2} \leq j \leq n$

# Analysis

$$T(n, m) = T(\lfloor \tfrac{n}{2} \rfloor, i) + T(\lceil \tfrac{n}{2} \rceil, m - i) + O(m)$$
$$\implies T(n, m) = O(m \log n)$$

# Application: Ciel and Gondolas[1]

给出 $n \times n$ 的矩阵 $U_{n \times n}$

把 $[0, n)$ 分成 $k$ 段 $[p_1 = 0, p_2), [p_2, p_3), \ldots, [p_k, p_{k+1} = n)$

使得 $\sum_{i=1}^{k} \sum_{p_i \le x < y < p_{i+1}} U_{x,y}$ 最小

$n \le 4000, k \le 800$

[1] Codeforces Round #190 E

# Analysis

记 $w(i,j) = \sum_{i \le x < y < j} U_{x,y}$

$w$ 满足四边形不等式

# Proof

# Analysis (Con'd)

设 $f_k(i)$ 表示将 $[0, i)$ 划分成 $k$ 段的最小代价

则 $f_k(i) = \min_{0 \le j < i} f_{k-1}(j) + w(j, i)$

# Analysis (Con'd)

$$f_{k+1}(i) = \min_{0 \le j < i} f_k(j) + w(j, i)$$

$$\text{设 } M(j, i) = f_k(i) + w(i, j)$$

$$M(c, a) + M(d, b) - M(c, b) - M(d, a)$$

$$= (f_k(a) + w(a, c)) + (f_k(b) + w(b, d)) - (f_k(b) + w(b, c)) -$$

$$(f_k(a) + w(a, d))$$

$$= w(a, c) + w(b, d) - w(b, c) - w(a, d) \le 0$$

# Analysis (Con'd)

$M$ 满足四边形不等式 $\implies$ $M$ 满足完全单调性

$f_{k+1}(i) = \min_j M(i, j)$

$O(n \log n)$ 求解 $f_{k+1}(\cdot)$

# Application: All-Pair Farthest Points[2]

给出 $n$ 个点的凸包 $p_1 p_2 \ldots p_n$

求 $D(i) = \max_j \|p_i p_j\|$

$n \leq 30,000$

[2] Rujia Liu's Present 4: A Contest Dedicated to Geometry and CG Lovers

# Analysis

令 $d(i,j) = \begin{cases} \|p_i p_{j \bmod n}\| & i < j < i+n \\ -\|p_i p_{j \bmod n}\| & \text{otherwise} \end{cases}$

则 $D(i) = \max_j d(i,j)$

# Analysis

$d$ 满足（反）四边形不等式

$$d(a,c)+d(b,d)\geq d(a,d)+d(b,c)$$

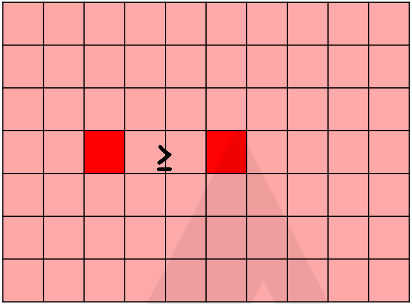
$O(n\log n)$ 求解 $D(\cdot)$

# $O(n)$ Row Minima

Aggarwal, Alok, et al. "Geometric applications of a matrix-searching algorithm." Algorithmica 2.1-4 (1987): 195-208.

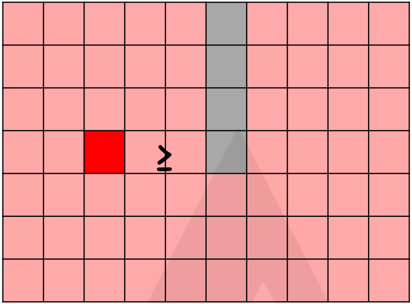
# The SMAWK Algorithm

Dead elements following a comparison:



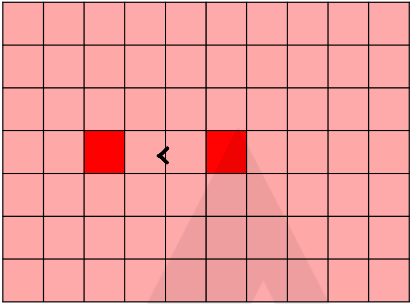
# The SMAWK Algorithm

Dead elements following a comparison:



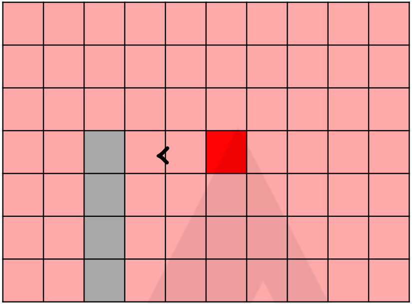
# The SMAWK Algorithm

Dead elements following a comparison:



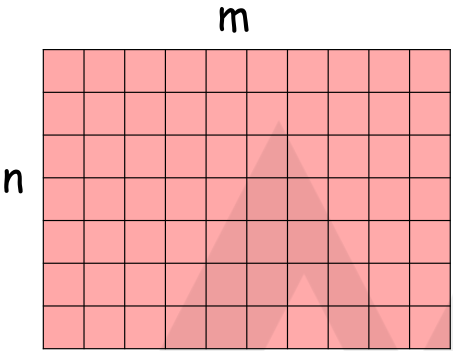
# The SMAWK Algorithm

Dead elements following a comparison:

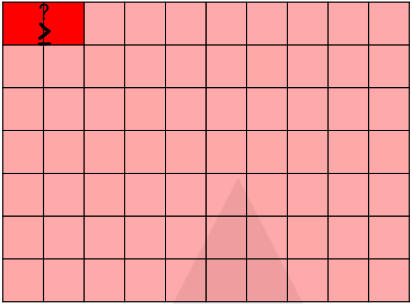


# The SMAWK Algorithm (reduce)

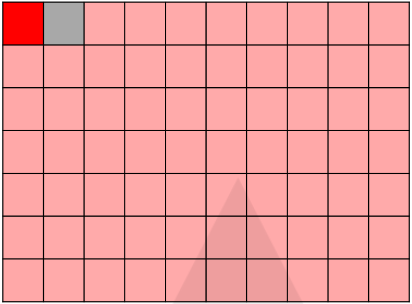
Step 1: Reduce the number of columns to n



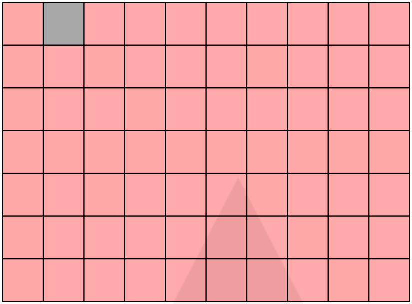
# The SMAWK Alg. (reduce)



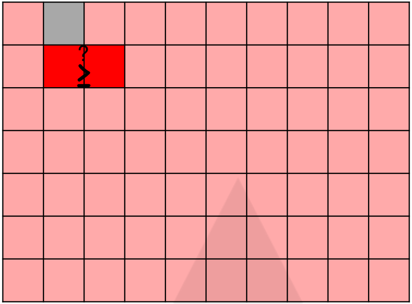
# The SMAWK Alg. (reduce)



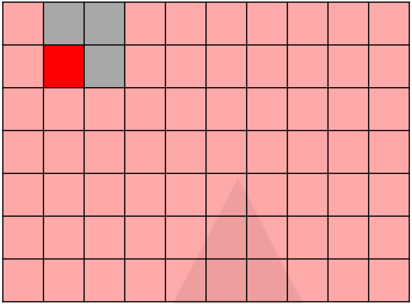
# The SMAWK Alg. (reduce)



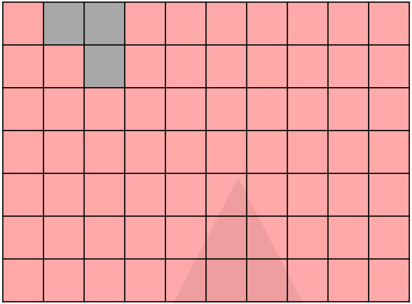
# The SMAWK Alg. (reduce)



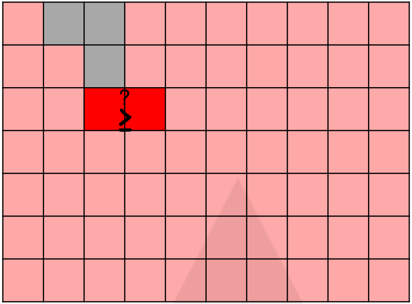
# The SMAWK Alg. (reduce)



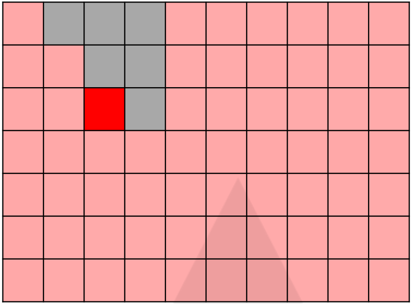
# The SMAWK Alg. (reduce)



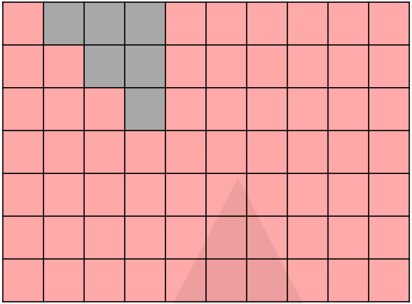
# The SMAWK Alg. (reduce)



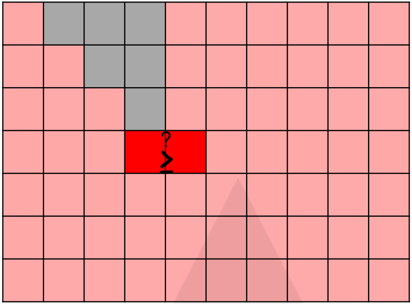
# The SMAWK Alg. (reduce)



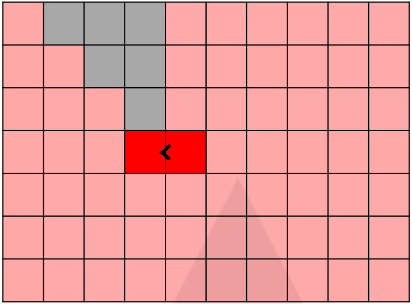
# The SMAWK Alg. (reduce)



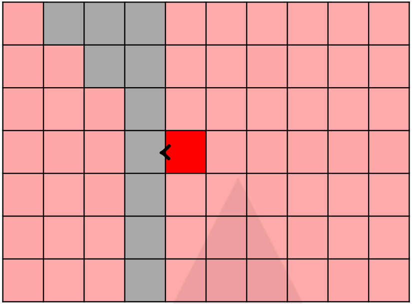
# The SMAWK Alg. (reduce)



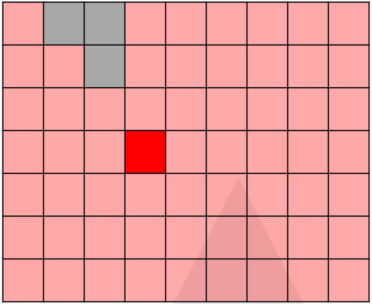
# The SMAWK Alg. (reduce)



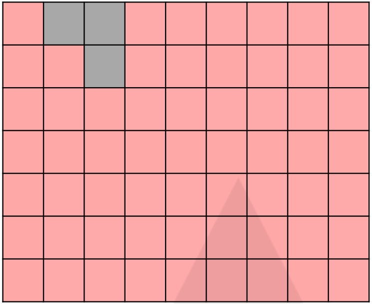
# The SMAWK Alg. (reduce)



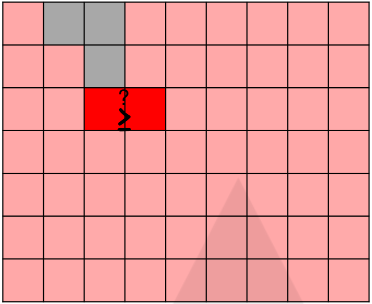
# The SMAWK Alg. (reduce)



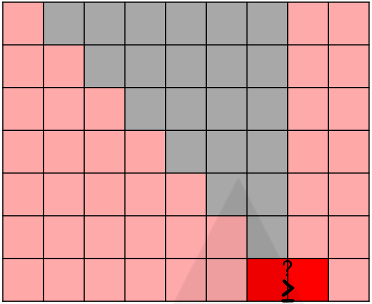
# The SMAWK Alg. (reduce)



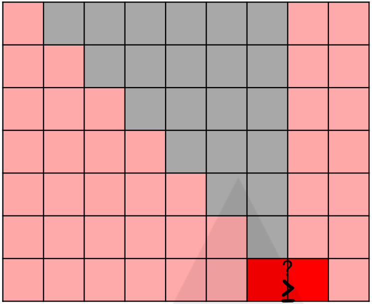
# The SMAWK Alg. (reduce)



# The SMAWK Alg. (reduce)

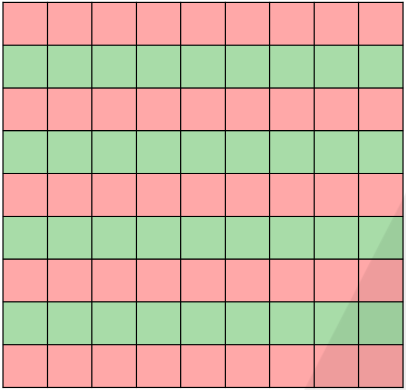


# Analysis of reduce

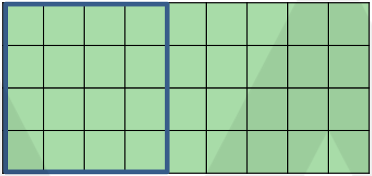



We go backwards only when a column is deleted ➔ (m-n) times ➔ go forward n + (m - n) times ➔ total 2m-n steps

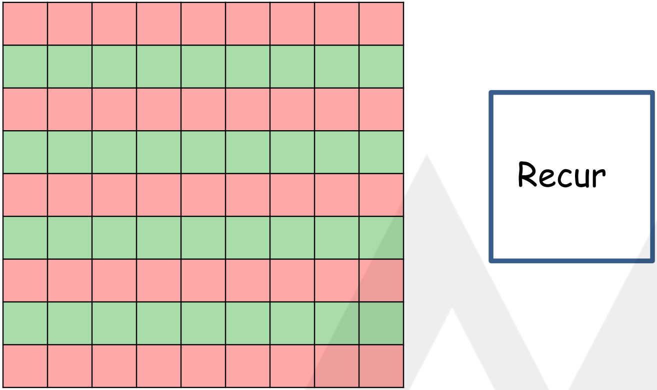
# The SMAWK Alg. (main recurrence)



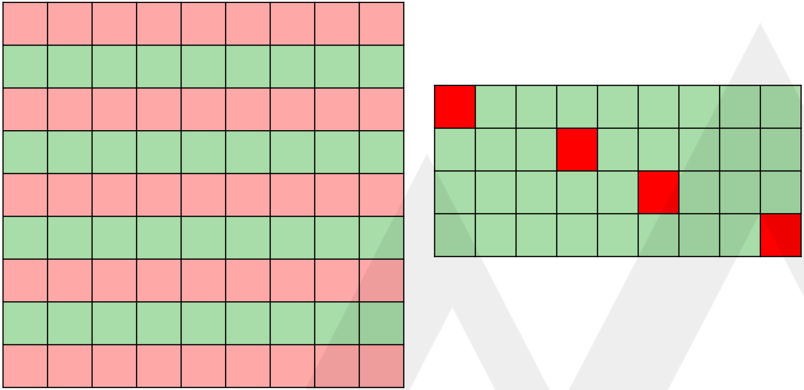
Reduce:



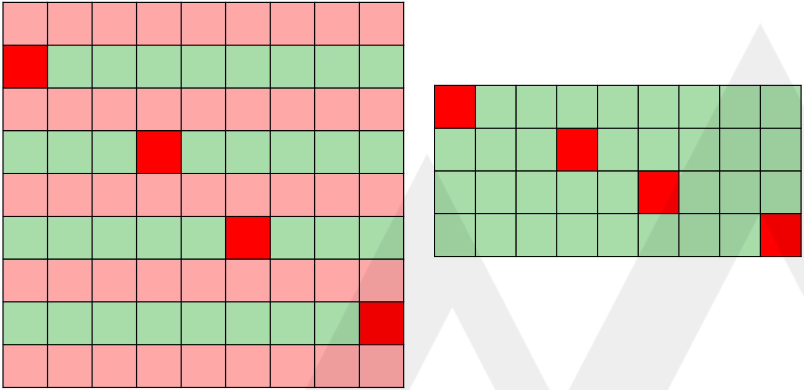
# The SMAWK Alg. (main recurrence)



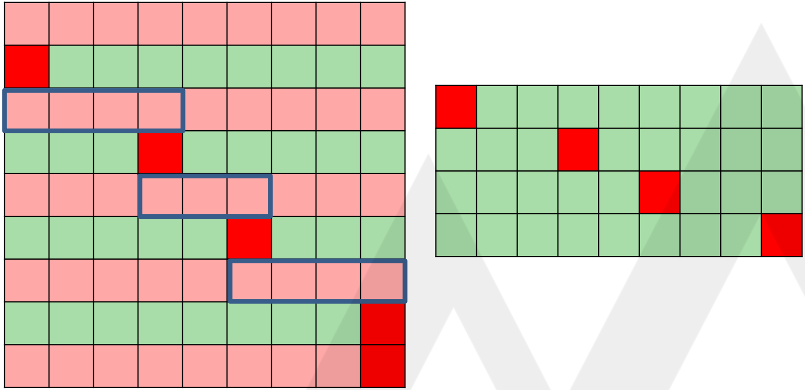
# The SMAWK Alg. (main recurrence)



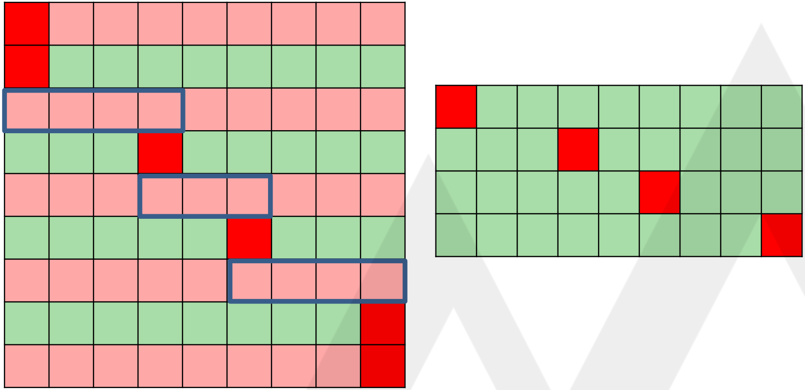
# The SMAWK Alg. (main recurrence)



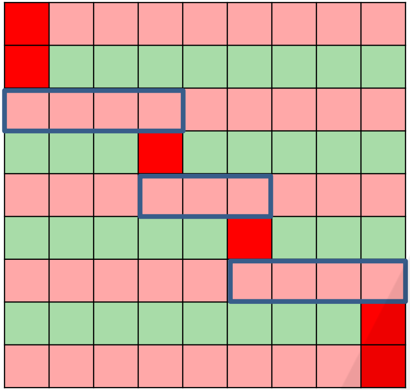
# The SMAWK Alg. (main recurrence)

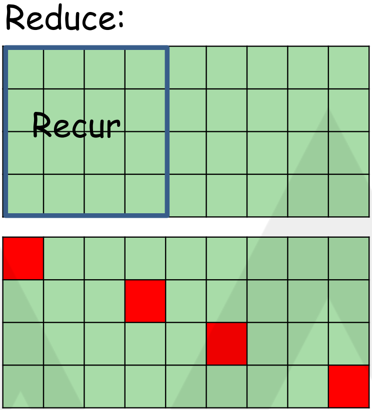


# The SMAWK Alg. (main recurrence)



# SMAWK (analysis)






$$T(n) = O(n) + T(n/2) = O(n) \qquad \square$$

# Online Row Minima

$$f(i) = \min_{j<i} f(j) + w(j, i)$$

$w$ 满足四边形不等式 $\implies$ $M(j, i) = f(j) + w(j, i)$ 满足完全单调性

无法使用 SMAWK

# Alternative Approaches


- Galil, Zvi, and Kunsoo Park. "A linear-time algorithm for concave one-dimensional dynamic programming." Information Processing Letters 33.6 (1990): 309-311.
- Larmore, Lawrence L., and Baruch Schieber. "On-line dynamic programming with applications to the prediction of RNA secondary structure." Journal of Algorithms 12.3 (1991): 490-515.

# Thanks!