Algorithms and Data Structures
for Faulty Memory

Computer Science Day 2008

Gerth Stølting Brodal

MADALGO
CENTER FOR MASSIVE DATA ALGORITHMICS

Department of Computer Science, University of Aarhus, June 20, 2008

MADALGO

CENTER FOR MASSIVE DATA ALGORITHMICS

Faculty

Post Docs

Administration

PhD students

Master Students

Programmers

www. madalgo.au.dk

MADALGO

CENTER FOR MASSIVE DATA ALGORITHMICS

August 18-21, 2008

MADALGO Summer School on Cache-Oblivious Algorithms

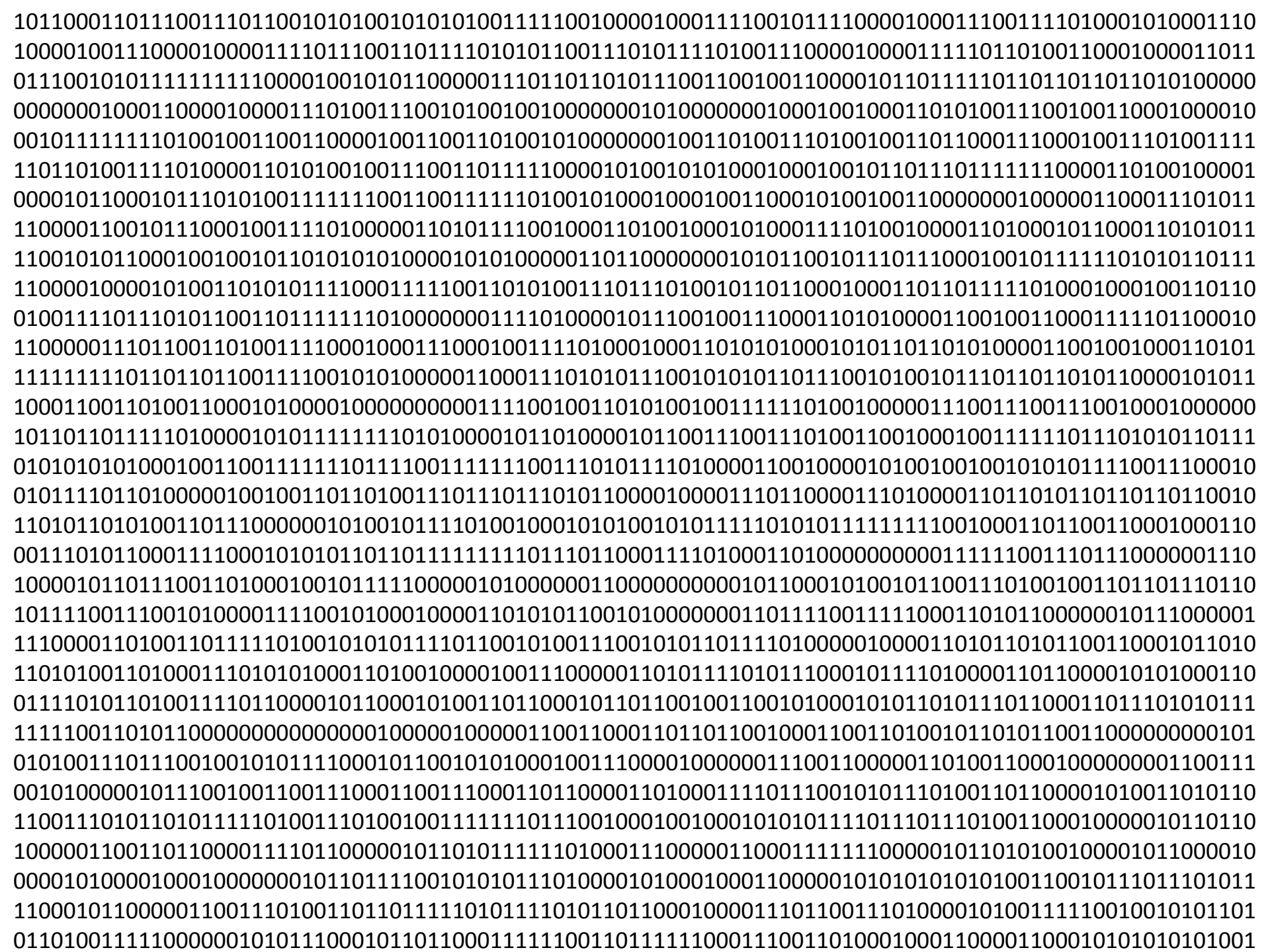
June 8-10, 2009

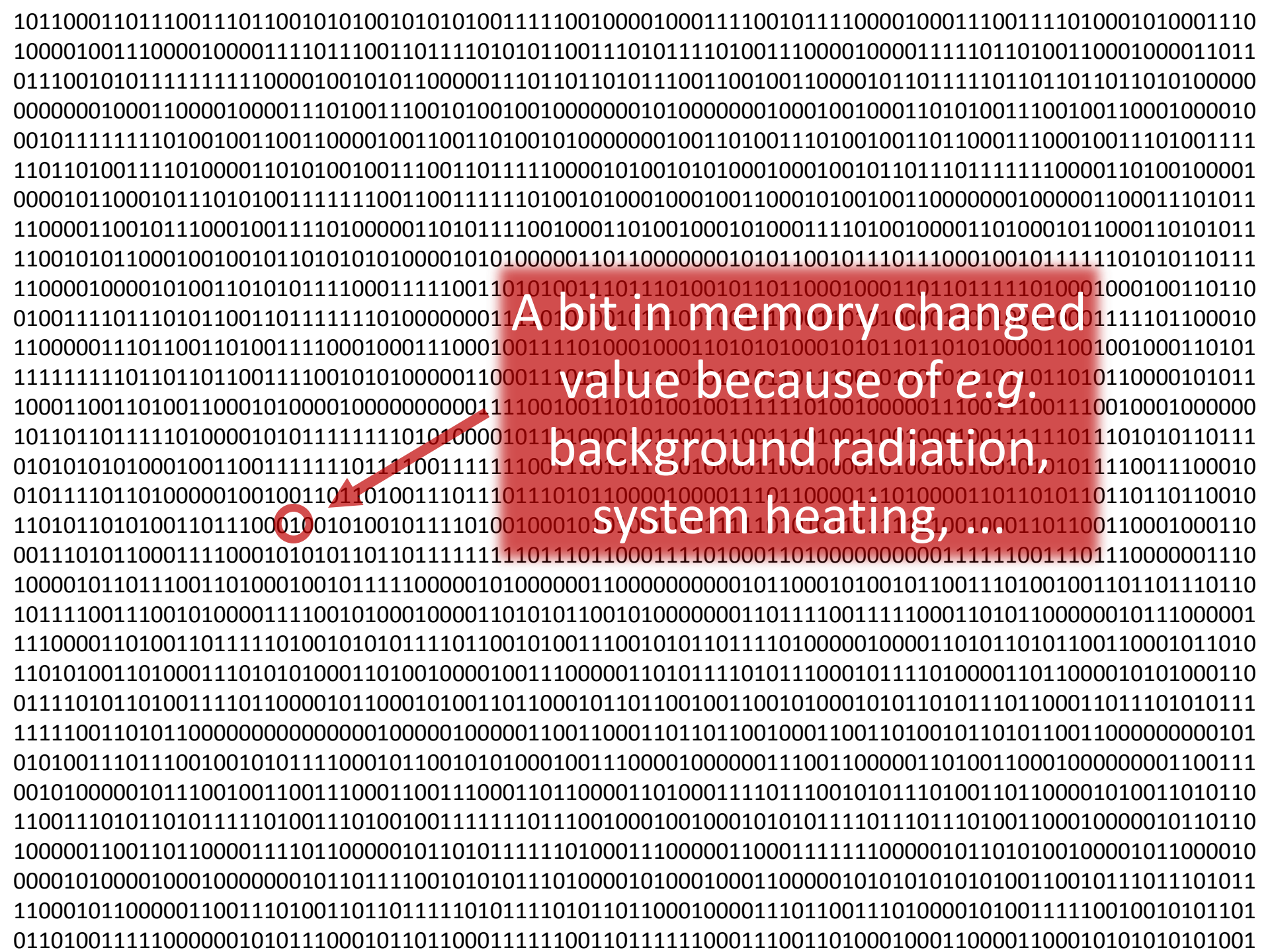
25th Annual ACM Symposium on Computational Geometry

www. madalgo.au.dk

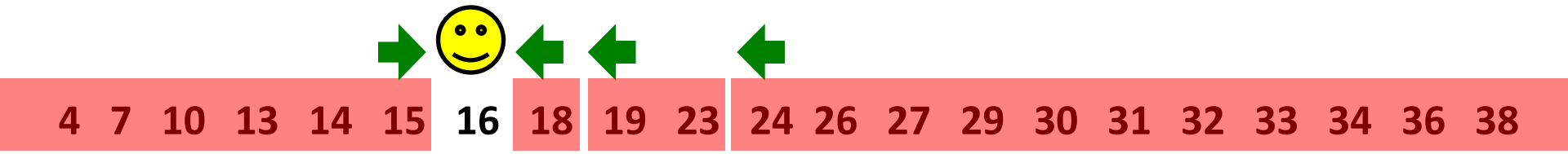
"You have to provide reliability on a software level. If you're running 10,000 machines, something is going to die every day."

— Google fellow Jeff Dean



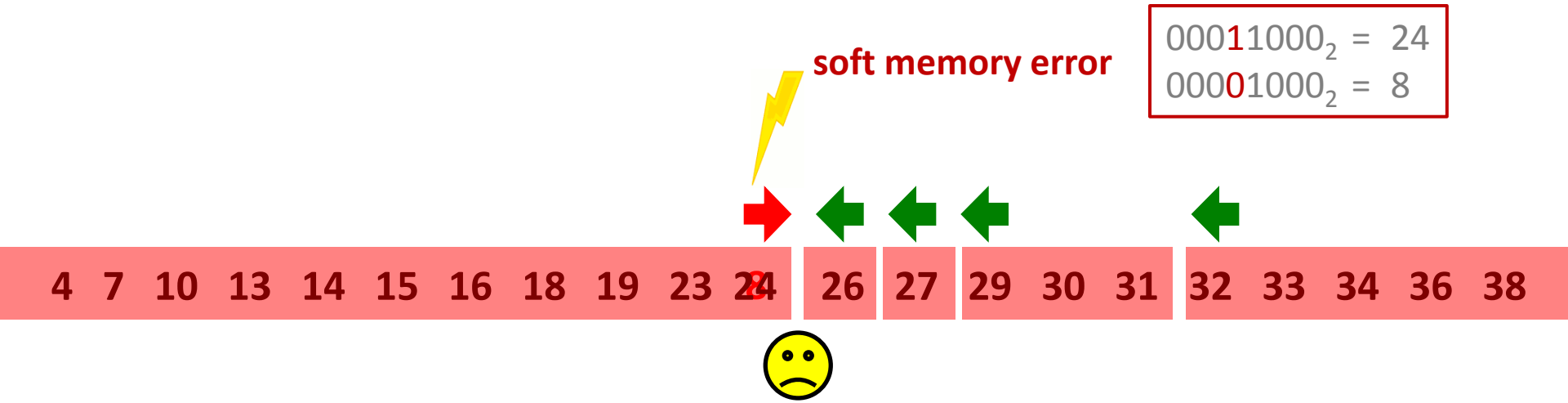

A bit in memory changed value because of *e.g.* background radiation, system heating, ...

Binary Search for 16



O(log *N*) comparisons

Binary Search for 16



Requirement: If the search key ocours in the array as an uncorrupted value, then we should report a match *!*

Where is Kurt ?

Where is Kurt ?

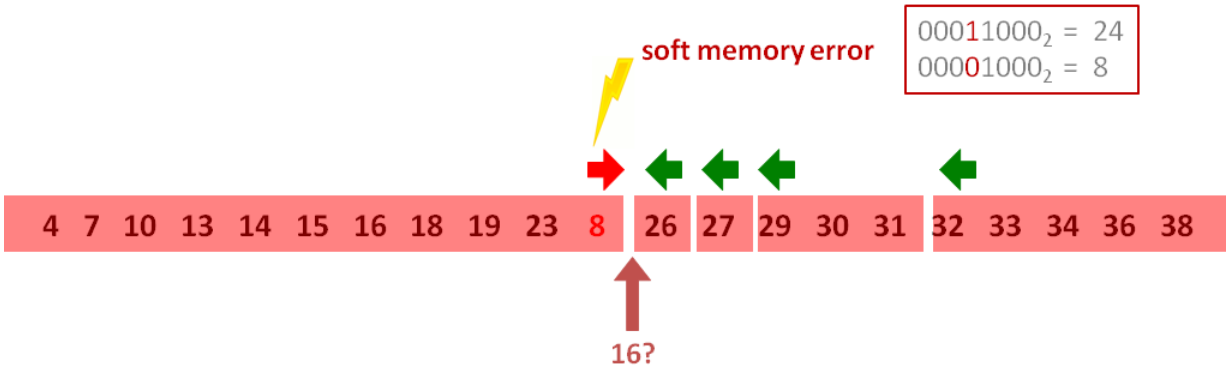
Where is Kurt ?

If at most 4 faulty answers then Kurt is somewhere here

Faulty-Memory RAM Model

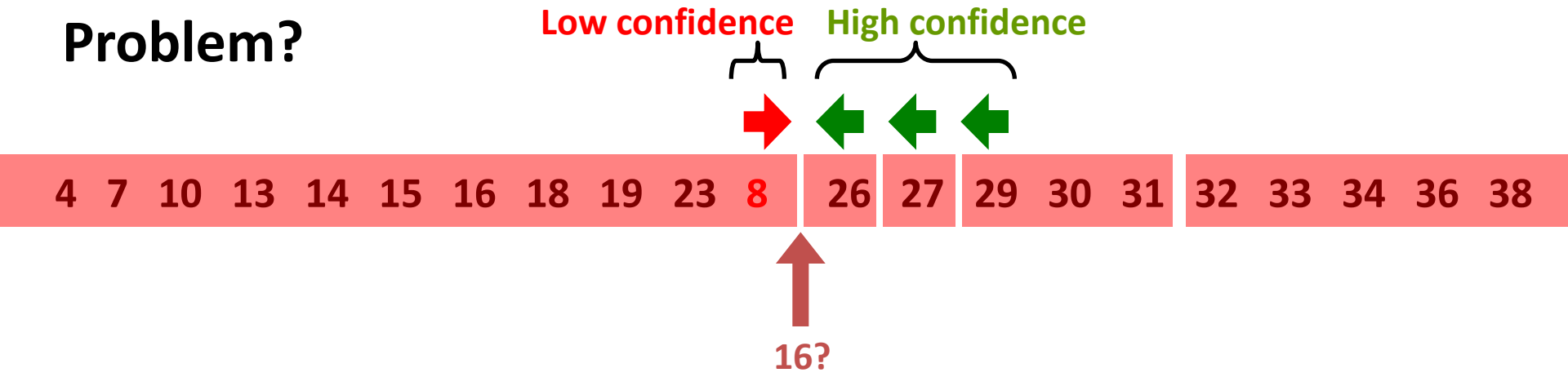
Finocchi and Italiano, STOC'04

- Content of memory cells can get corrupted
- Corrupted and uncorrupted content cannot be distinguished
- *O*(1) safe registers
- **Assumption**: At most δ corruptions



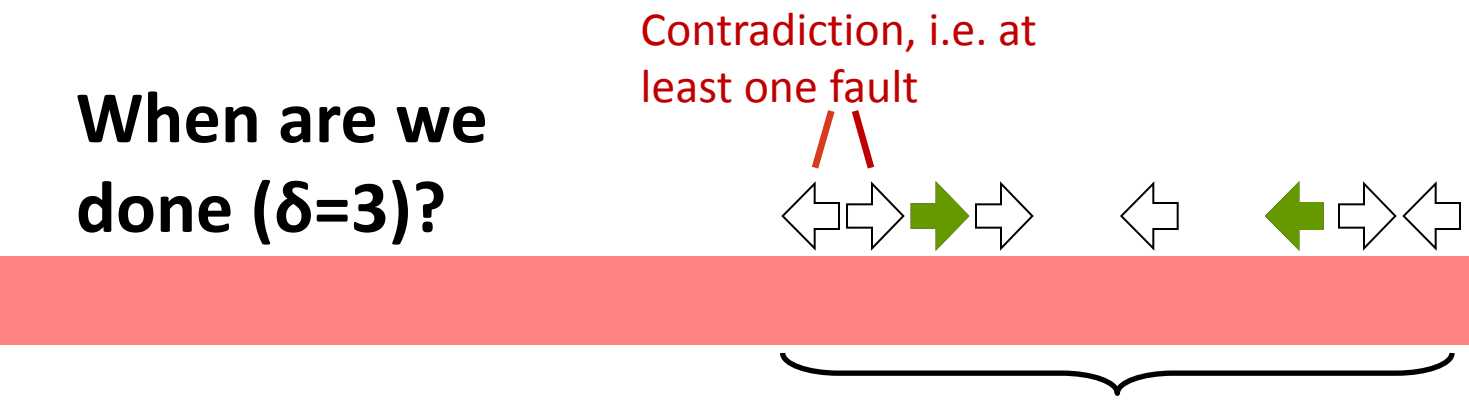
Faulty-Memory RAM: Searching

Problem?



Faulty-Memory RAM: Searching

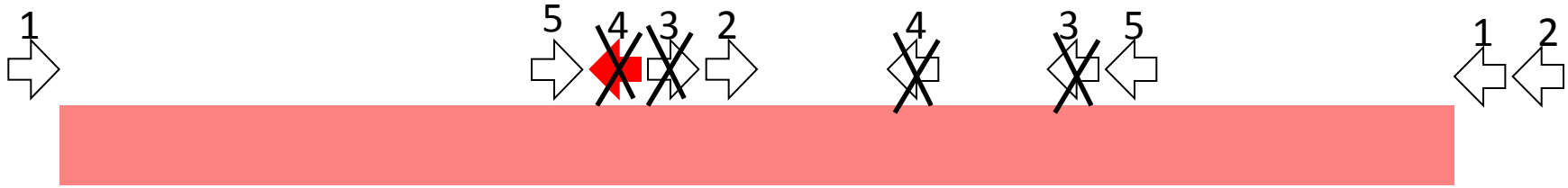
When are we done (δ=3)?



If range contains at least δ+1 ⇨ and δ+1 ⇦ then there is at least one uncorrupted ➡ and ⬅, i.e. *x* must be contained in the range

Faulty-Memory RAM: Θ(log *N* + δ) Searching

Brodal, Fagerberg, Finocchi, Grandoni, Italiano, Jørgensen, Moruz, Mølhave, ESA'07



If verification fails

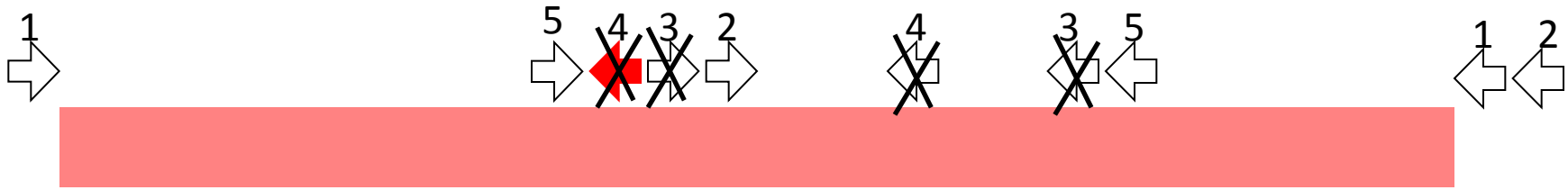
→ contradiction, i.e. ≥1 memory-fault

→ ignore 4 last comparisons

→ backtrack one level of search

Faulty-Memory RAM: Θ(log *N* + δ) Searching

Brodal, Fagerberg, Finocchi, Grandoni, Italiano, Jørgensen, Moruz, Mølhave, ESA'07

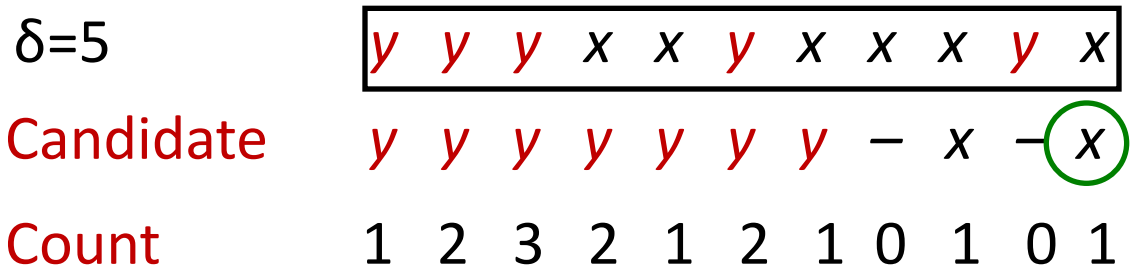


- Standard binary search + verification steps
- At most δ verification steps can fail/backtrack
- **Detail**: Avoid repeated comparison with the same (wrong) element by grouping elements into blocks of size $O(\delta)$

Faulty-Memory RAM: Reliable Values

- Store 2δ+1 copies of value x - at most δ copies uncorrupted
- x = majority
- Time $O(\delta)$ using two safe registers (candidate and count)

Boyer and Moore '91

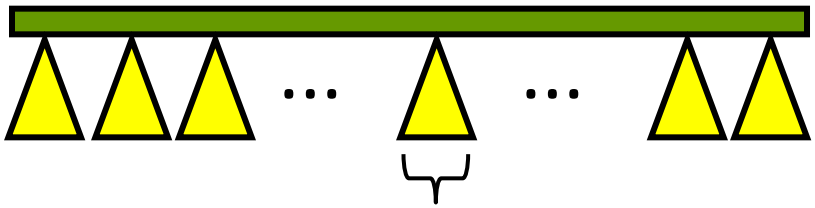


Faulty-Memory RAM: Dynamic Dictionaries

Brodal, Fagerberg, Fnocchi, Grandoni, Italiano, Jørgensen, Moruz, Mølhave, ESA'07

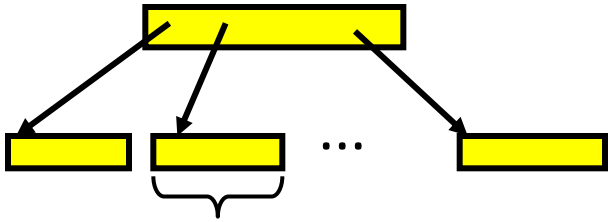
- Packed array
- Reliable pointers and keys
- Updates $O(\delta \cdot \log^2 N)$
- Searches = fault tolerant $O(\log N+\delta)$

Itai, Konheim, Rodeh, 1981



$\Theta(\delta \cdot \log N)$ elements

- 2-level buckets of size $O(\delta \cdot \log N)$
- Root: Reliable pointers and keys
- Bucket search/update amortized $O(\log N+\delta)$



$\Theta(\delta)$ elements

- **Search and update amortized $O(\log N+\delta)$**

Fault-Tolerant Results

- Merging, time $\Theta(N+\delta^2)$ — Finocchi, Grandoni, Italiano, ICALP'06
- Priority queue, time $\Theta(\log N+\delta)$ — Jørgensen, Moruz, Mølhave, WADS'07
- Sorting, time $\Theta(N \cdot \log N+\delta^2)$ — Finocchi, Grandoni, Italiano, ICALP'06
- Static and dynamic dictionary, time $\Theta(\log N+\delta)$
  Brodal, Fagerberg, Finocchi, Grandoni, Italiano, Jørgensen, Moruz, Mølhave, ESA'07
  Finocchi, Grandoni, Italiano, ICALP'06
- External-memory fault tolerant searching, $\Theta\left(\frac{1}{\varepsilon}\log_B N+\frac{\delta}{B^{1-\varepsilon}}\right)$ I/Os
  Brodal, Jørgensen, Mølhave, Submitted

Allan G. Jørgensen and Gerth S. Brodal at MFCS'07