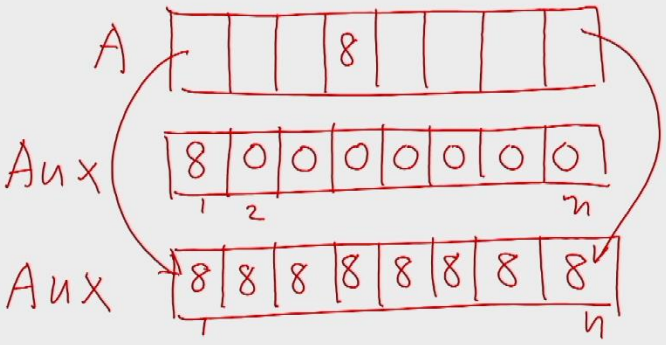
Airplane mode off

Applications of Prefix Sums

- Compaction
- Broadcast
- Prefix Minima/Maxima
- Rightmost element to the left

Broadcast



$O(\log n)$ time

$\frac{n}{\log n}$ procrs

EREW PRAM

Prefix / Suffix Minima / Maxima

4	3	1	9	8	6	2	7	
4	4	4	9	9	9	9	9	→ max
4	3	1	1	1	1	1	1	→ min
9	9	9	9	8	7	7	7	← max
1	1	1	2	2	2	2	7	← min

Sum → min / max

$O(\log n)$ time

$n / \log n$ processors

on an EREW PRAM

An array in which only some elements are useful

$-4--3--8-9-$

at each position, the right most element to the left of that

(i_1, j_1) and (i_2, j_2)

$(i_1, j_1) > (i_2, j_2)$

if $i_1 > i_2$

or $i_1 = i_2$ and $j_1 > j_2$

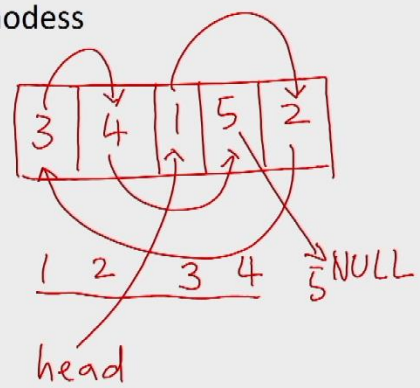
Pointer Jumping

Linked List in an Array

next/ parent

- Physical/Logical Order of nodess

need not be the same!

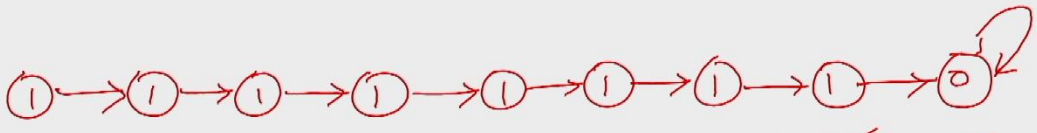


Array of size n

n processors.

1 processor per node

CREW PRAM $\quad n = 2^k$



Ranks from the right

start with 0

nodes at a distance of

at most 1 already

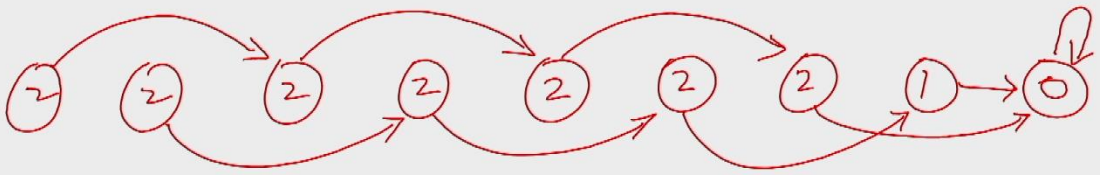
have the correct ranks

1st step

every node updates its rank

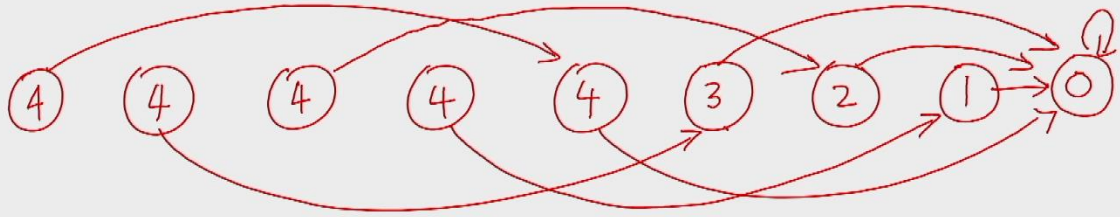
rank[i] : rank[i] + rank[p[i]]

p[i] : p[p[i]]

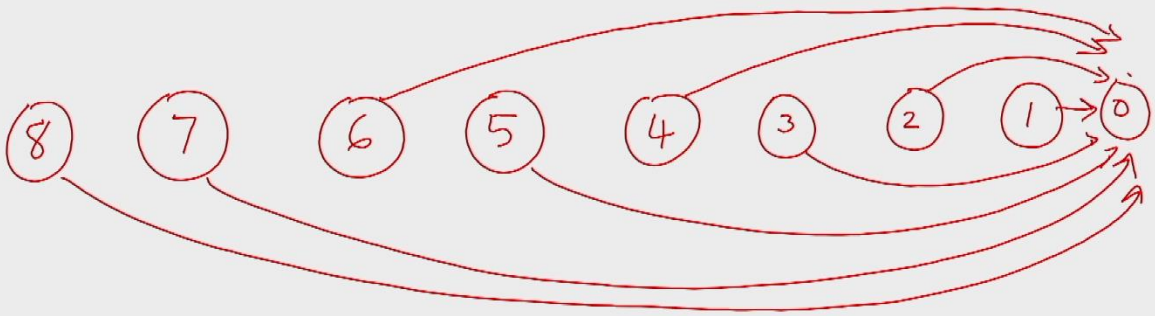


All nodes at a dist. of ≤ 2 from the right most node have the correct ranks

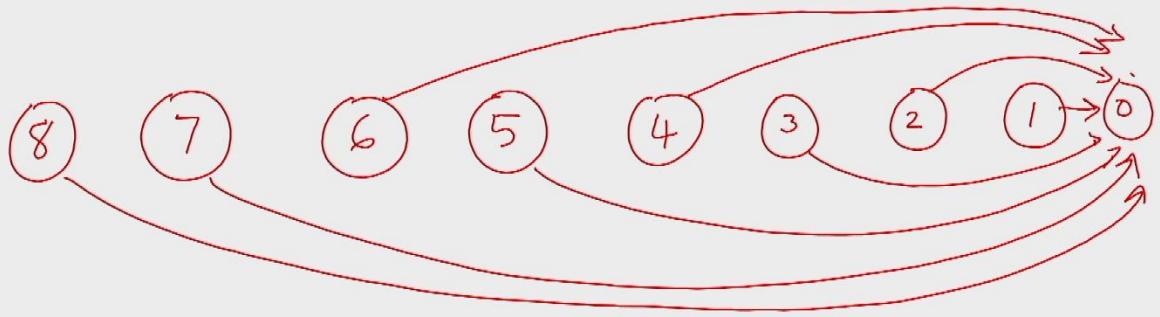
1 unit of time



Nodes at dist ≤ 4 have the correct rank



Ranked from the Right end !



Ranked from the Right end !

subtract rank from |array|

1 2 3 4 5 6 7 8 9

$O(\log n)$ time using

n processors

CREW PRAM

$O(n \log n)$ cost

non optimal

List Ranking

Input: A linked list represented as an array $s[1 \ldots n]$ of pointers; $s[i]$ is the successor of node i of the list; if node i is the last in the list, $s[i] = i$.

Output: An array $r[1 \ldots n]$ of "ranks", the number of pointers to be traversed in going from node i to the last node in the list. Model: EREW PRAM.

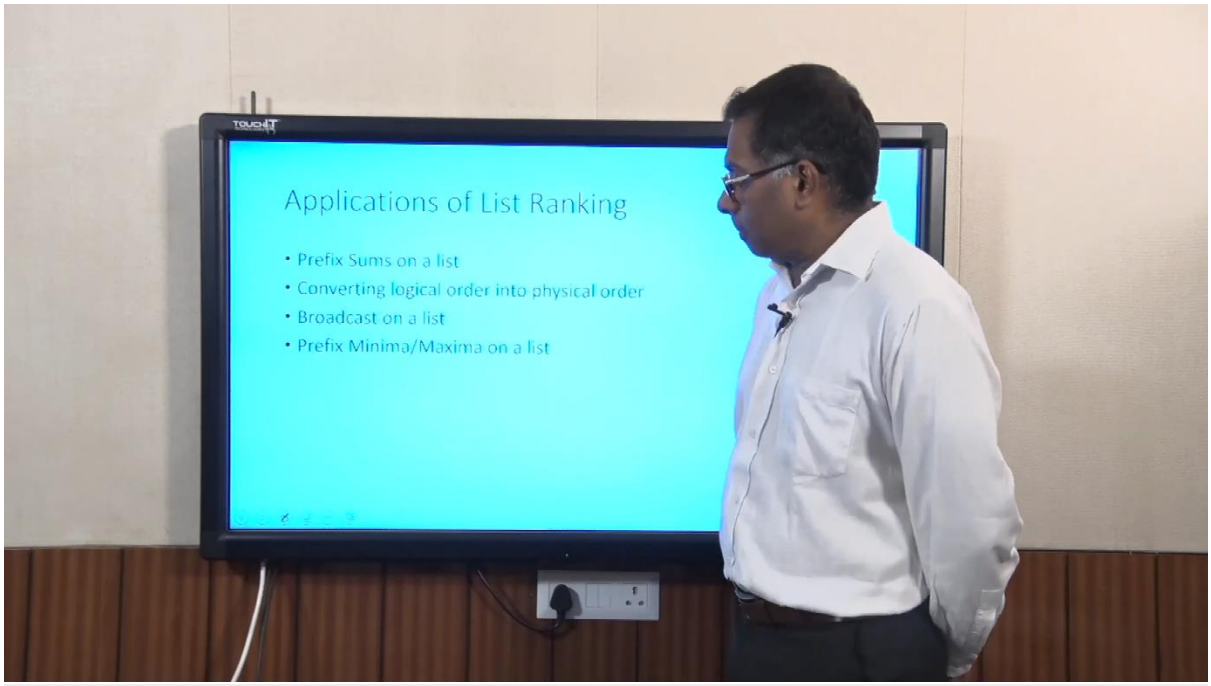
```
{
    pardo for 1 ≤ I ≤ n
        if (s[I] == I) r[I] = 0;
        else r[I] = 1;
```

List Ranking (Cont'd)

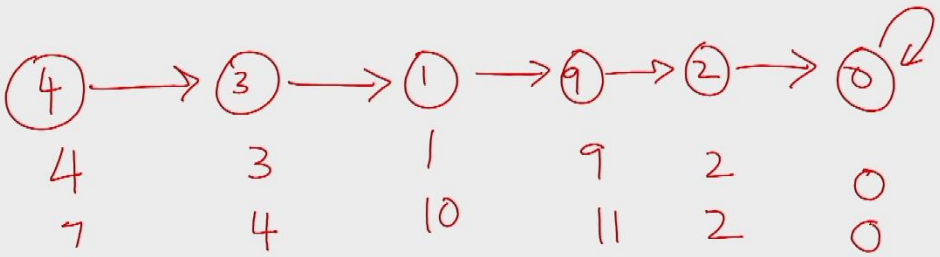
```
    for s = 1 to log n do
        pardo for 1 ≤ I ≤ n
            if (s[I]! = s[s[I]])
            {
                r[I]+ = r[s[I]];
                s[I] = s[s[I]];
            }
    return r;
}
```

How is the read conflict resolved?

Hint: A minor modification will do.



Prefix sums on a list

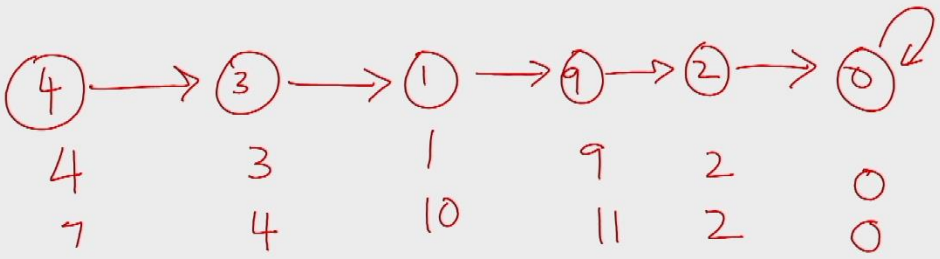


Size n

- nodes of dist ≤ 1 ✓
- nodes of dist ≤ 2 ✓
- ——"—— ≤ 4 ✓
- ——"—— $\leq 2^k$ ✓

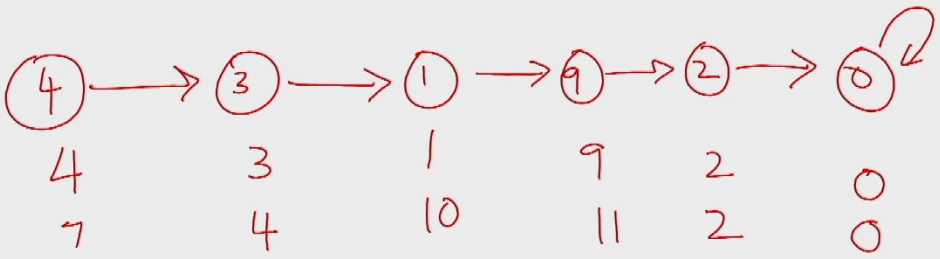
$\log n$ steps → solved!

Prefix sums on a list



$O(\log n)$ time $O(n \log n)$ Cost

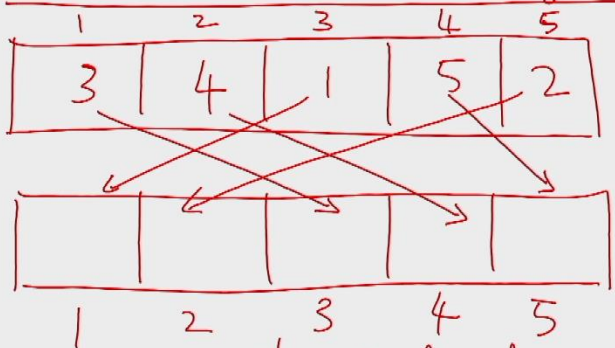
Prefix sums on a list



$O(\log n)$ time $O(n \log n)$ Cost

EREW PRAM

Logical Order to Physical Order

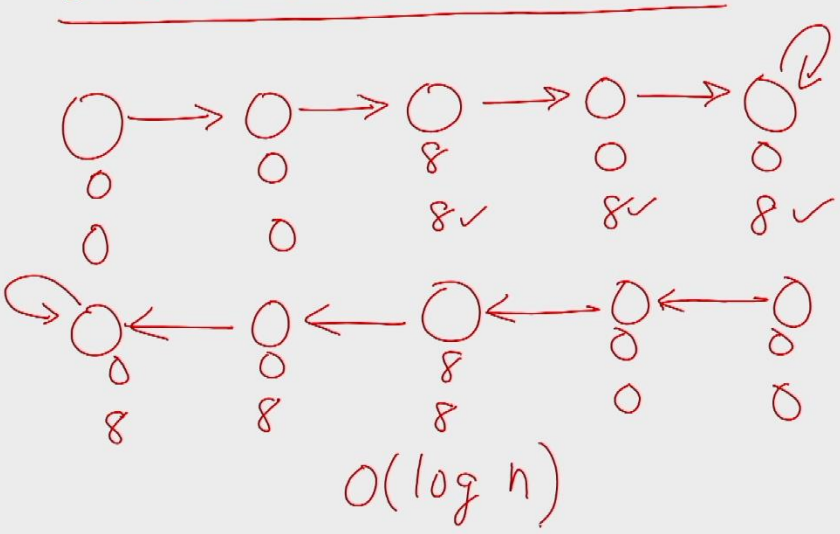


List Ranking

copy

physical order . logical order

Broadcast on a list



$O(\log n)$