Let $\oplus$ be an associative binary op. e.g., $\oplus$, $\cdot$.

**Def.** All-Prefix-Sums

Input $[a_0, \ldots, a_{n-1}]$

Output $[a_0, a_0 \oplus a_1, a_0 \oplus a_1 \oplus a_2, \ldots, a_0 \oplus \cdots \oplus a_{n-1}]$

**Def.** Prescan

Output $[I, a_0, a_0 \oplus a_1, \ldots, a_0 \oplus \cdots \oplus a_{n-1}]$

**Note** Prefix Sum $(A) = \text{Prescan}(A) \oplus A$

cellwise addition
Prescan Application

Packing Out Memory

<table>
<thead>
<tr>
<th>Location</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contents</td>
<td>φ *</td>
<td>φ</td>
<td>φ</td>
<td>* *</td>
<td>φ</td>
<td>*</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Location</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contents</td>
<td>* * * *</td>
<td>φ</td>
<td>φ</td>
<td>φ</td>
<td>φ</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Use Prescan

Input: (0,1,0,0,1,1,0,1)
Prescan: (0,0,1,1,1,2,3,3)

Note: Google: MapReduce not defined for arrays.
Parallel Precan By Example

Input \( (3, 1, 7, 0, 4, 1, 6, 3) \) \( \oplus \) addition

Precan Alg 1) Compute tree of partial sums.
   2) Set root to zero
   3) Run Down!

![Diagram]

Proc Down: Inll do
   a) Right-Child \( \leftarrow \) Parent \( \oplus \) left-child
   b) Left-Child \( \leftarrow \) Parent
   c) Repeat on children
Down

\[ \begin{array}{c}
\text{Precan} (0, 3, 4, 11, 11, 15, 16, 22) \\
\text{Input} (3, 1, 7, 0, 4, 1, 6, 3)
\end{array} \]

\[ T(n) = O(\log n) \]

\[ P(n) = O(n) \quad \Rightarrow \quad P \cdot T = O(n \cdot \log n) \]
List Ranking

Input: linked-list

Output: A mark on each node s.t.
mark = distance from head on
= distance to tail

Head 0 ——> 1 ——> 2 ——> 3 ——> 4
Tail 4 ——> 3 ——> 2 ——> 1 ——> 0

Assume:
1) pointers are in consecutive memory
2) we know location of head & tail
3) pointers in arbitrary order.
Wyllie's Alg

Inll rank(\(\text{!}\)) := 1 \text{ ; } \text{rank}(\text{tail}) = 0
Inll while succ(head) \neq \text{nil} do
  \text{if succ (\(\text{!}\)) \neq \text{nil} do}
    \text{rank (\(\text{!}\)) := rank (\(\text{!}\)) + rank (succ (\(\text{!}\)))}
    \text{succ (\(\text{!}\)) := succ (succ (\(\text{!}\)))}
  fi

\# processors = n
Time = O(hgn)
Work = O(h^2 gn)
CREW model

Goal
O(h^2 gn)

O(hn)
Wyllie's Alg

Initial

After one round

After two rounds
Prescan on a linked list

Recursively splice out every other element!

Fill in missing node values.
Random - Mate

1) Contraction Phase

1) Each live node randomly picks a sex

2) If $F \rightarrow a \rightarrow M \rightarrow b \rightarrow X$ then $a + b$

3) Step when head points to nil.

(Only head is live)
In the expansion phase we run contraction phase "backward".

Eg before

\[
\begin{array}{c}
\text{live} & b & \text{dead} & \text{live} \\
V & ? & W
\end{array}
\]

\[
\begin{array}{c}
\text{live} & b & \text{live} & \text{live} \\
V & V+b & W
\end{array}
\]
**Random Mate Timing Analysis**

**Thm** The contraction phase for Random Mate terminates with high prob.

**Pr** let \( P_i^k \) = Event that node \( i \) is still live after \( k \) rounds.

**Note** If node \( i \) not head then \( \Pr[P_i^k] = \frac{1}{4} \)

Since each round is independent

\[
\Pr[P_i^k] = \left(\frac{3}{4}\right)^k 
\]

for node \( i \) not head.

Set \( k = \lceil \frac{1}{c} \log_{4/3} n \rceil \)

\[
\Pr[P_i^k] = \left(\frac{1}{4/3}\right)^k \leq \frac{1}{(4/3)^{\lceil \frac{1}{c} \log_{4/3} n \rceil}} = \frac{1}{n^c}
\]

**Let** \( P^k \) = Event that some non-head node is still live.

**Assume** that node 0 is the head.

\[ P^k = P_1^k \cup \ldots \cup P_n^k \]

\[
\Pr[P^k] = \Pr[P_1^k \cup \ldots \cup P_n^k] 
\]

\[
\leq \Pr[P_1^k] + \ldots + \Pr[P_n^k] \leq \frac{n}{n^c} = \frac{1}{n^{c-1}}
\]

\( \square \)
Euler Tours

Problem: Preorder number a tree
Input: A tree stored using pointers
Output: Preorder numbering

Weighted list Ranking

Input:

Output:

Preorder

Use First-Visit value.
Preorder
1 = weight of down edge
0 = weight of up edge

Inorder

Use "Middle-Visit" Value

Postorder?

Size of Subtree?