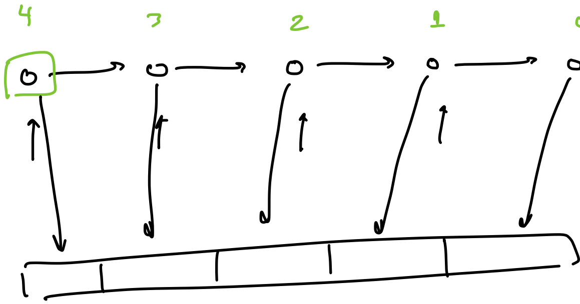
Today: pointer-manipulation / tree D.S.

lists, trees, graphs



collection of linked lists

v (1) what list is v in?

o (2) where is it in the list?

- Euler Tour Technique
- BW-Decoding

] Apps of today's lecture

Tree Contraction (next Tuesday)

LCAs, RMQ (Th)

List Ranking - lot of ink 1980s

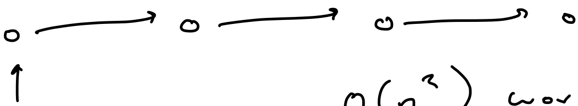
Input: collection of nodes in potentially multiple linked lists (input = pointers to the nodes)

Output: Compute the position of each node in the list it belongs to.

Very easy to do in $O(n)$ time sequentially.

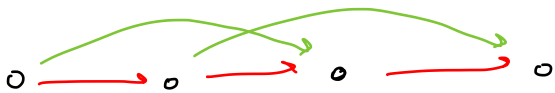
Wyllie's (deterministic) Algorithm:

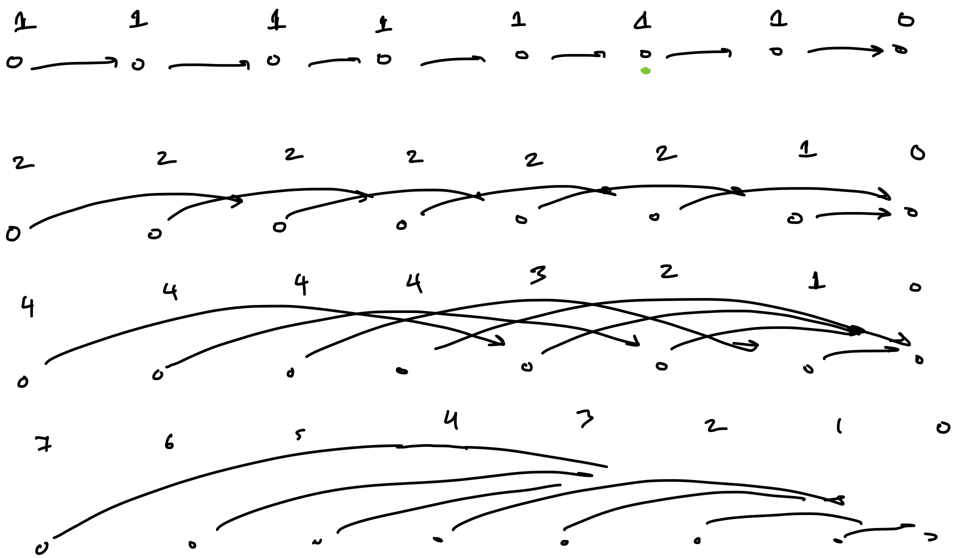
- "pointer jumping" technique



$O(n^2)$ work

$O(n)$ depth





$$D[i] = D[i] + D[succ[i]]$$

$D[i] = 1$ if $succ(i) \,!= \perp$, 0 o.w.

succ, succ'

for $k = 1$ to $\log(n)$:

    parfor i in $[1:n]$:

4	$\perp$	2	3
1	2	3	4

        $D[i] = D[i] + D[succ[i]]$

        $succ'[i] = succ[succ[i]]$

    swap(succ, succ')

After the h-th iteration:

(1) If $succ[i] \neq \perp$, $D[i] = 2^h$ and there are 2^k edges to $succ[i]$ in the list

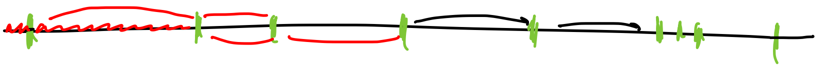
(2) If $succ(i) = \perp$, then $D(i) =$ rank(i)

$\Rightarrow$ total # rounds is $\log(n)$

Work $= O(n \log(n))$

Depth: $O(\log^2 n)$ (in BF model).

$O(n)$ work overall



$O(n\log(n))$ work $\quad \frac{n}{\log(n)} \Rightarrow O(n)$ work overall.

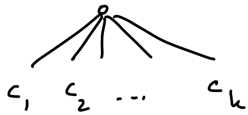
Random sampling with rate $\frac{1}{\log(n)} \Rightarrow \frac{n}{\log(n)}$ samples in expectation.

$$P[\text{Bad Event}] \le \frac{1}{n^c} \Rightarrow P[\text{Good Event}] \ge 1 - \frac{1}{n^c}$$

We say that an event (or cost bound) occurs (holds) whp <u>with high probability</u> if

$g(n) \in O(f(n))$ whp if $g(n) \in O(c \cdot f(n))$ with probability $\ge 1 - \frac{1}{n^c}$

$$\begin{array}{ll} c_1 & E[c_1] \\ c_2 & E[c_2] \\ \vdots & \vdots \\ c_k & E[c_k] \end{array} \qquad E[Alg] = \sum_i E[c_i]$$



$D(Alg) = \max_i D(i)$

Message: need to bound the prob. that every component doesn't finish within some span bound.

$$P(\text{bad event}] \le \frac{1}{n^c}$$

$$\le \frac{1}{n^{c-2}}$$

Boole's inequality / Union-bound. (for n^2 components).

Chernoff Bound:

$$P(E_1 \cup E_2 \cup \dots \cup E_n] \le P[E_1] + P[E_2] + \dots + P[E_n]$$

Let $X_1 \dots X_n$ be independent R.V.'s

where $X_i \in [0,1]$

Define $X = \sum_{i=1}^{n} X_i$, $p_i = E[X_i]$, $\mu = E[X]$

$\forall\, \delta \ge 0$:

$$\Pr[X \ge (1+\delta)\mu] \le \exp(-\delta^2\mu/(2+\delta))$$

- Hoeffding's inequality, Bernstein's inequality

(1) $O\left(\frac{n}{\log(n)}\right)$ samples whp

(2) Want to bound the search length $\forall$ sample

---

(1) $X_i = $ I.R.V. (indicator r.v.) $= 1$ iff i-th node is sampled

$$X = \sum_{i=1}^{n} X_i \quad , \quad P[X_i = 1] = \frac{1}{\log(n)}$$

$$E[X] = \frac{n}{\log(n)} \in \Omega(\log(n))$$

Apply Chernoff:

$$P(X \geq (1+\delta)E[X]) \leq \exp\left(-\frac{\delta^2 E[X]}{3}\right)$$

$$\leq \exp(-\delta^2 \log(n)/3)$$

$$= \frac{1}{n^{\delta^2/3}} \approx \frac{1}{n^c}$$

$\Rightarrow O\left(\frac{n}{\log(n)}\right)$ samples whp.

---

(2) Argue that traversal length for an arbitrary sample is short: $O(\log^2 n)$ whp.

$\mathrm{Geo}\left(\frac{1}{\log(n)}\right)$

Z = # vertices traversed by our sample

$$P(Z \geq x) = \left(1 - \frac{1}{\log(n)}\right)^{x}$$

$$x = c\log^2(n)$$

$$P(Z \geq c\log^2(n)) = \left(1 - \frac{1}{\log(n)}\right)^{c\log(n)\log(n)}$$

$$1+x \leq \left(1 + \frac{x}{n}\right)^{n} \leq e^{x}$$

$$n \geq 1, \quad |x| \leq n$$

$$\leq e^{-c\log(n)}$$

$$= \frac{1}{n^c}$$

$\Rightarrow$ high probability bound for one sample.

TODO: is this tight?

What about all samples? Boole's inequality

union bound

$$P[E_1 \cup E_2 \cup \ \cdots \ \cup E_n] \leq P[E_1] + P[E_2] + \cdots + P[E_n]$$

(1) $O\left(\frac{n}{\log(n)}\right)$ samples whp.

$$P(\text{any sample has traversed length} \geq c\log^2(n)) \leq n\left(\frac{1}{n^c}\right) = \frac{1}{n^{c-1}}$$

(2) Each one traverses $O(\log^2 n)$ nodes whp.

Randomized List Rank(A)

(1) Samples = sample every node w.p. $\frac{1}{\log(n)}$

(2) Splice out all non-sampled nodes from the list

(3) $\underbrace{\text{Determenitic List Rank}}_{\text{wyllie}}$ (Samples)

(1) $O(n)$ work, $O(\log(n))$ depth

(2) $O(n)$ work, $O(\log^2 n)$ depth whp

(3) $O(n)$ work whp. and $O(\log^2 n)$ depth (det).

$O(n)$ work whp, $O(\log^2 n)$ depth whp.

Historical Notes:

- Cole-Vishkin : $O(n)$ work $O(\log n \log^*(n))$ time algorithm.

  "Ruling Sets"

  $O(\log^2(n) \log^*(n))$ depth alg.

  May discuss in a future lecture