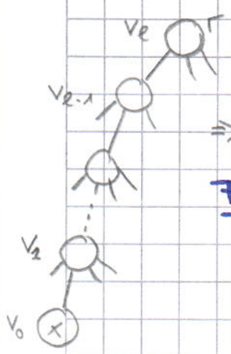
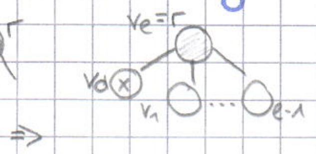
## 5) Verbesserung 4)

weighted union + Pfad Komprimierung (path. compression)

→ Find(x) durchläuft einen Pfad $v_0, \dots, v_e$ von Knoten wobei $v_0 = x$, $v_e$ = Wurzel r

<u>Idee</u>: Hänge $v_0, \dots, v_{e-1}$ direkt an die Wurzel r




<u>Find(x)</u>

```
r ← x;                              // r ist Wurzel
while vater[r] ≠ 0 do
        r ← vater[r];
od
                 (≠ r)
while vater[x] ≠ 0 do
      p ← vater[x];
      vater[x] ← r;
      x ← p;
od
```

<u>Beobachtung</u>:

<u>Pfadkompremierung</u>:

- verteuert aktuelle Find aber nur um konstanten Faktor → $O(\log n)$
- macht spätere Finds erheblich billiger

<u>Satz</u> (Tarjan)

Bei Union-Find mit weighted Union und path compression hat eine beliebige Folge von $n-1$ Unions und $m \geq n$ Finds die Gesamtkosten $O(m \cdot \alpha(m,n))$

Eine Variante der Ackermann-Funktion und ihre Inverse

$A : \mathbb{N}_0 \times \mathbb{N}_0 \rightarrow \mathbb{N}_0$

$$\left.\begin{array}{ll} A(i,0) = 0 & \text{für } i \geq 0 \\ A(0,x) = 2x & \text{für } x \geq 0 \\ A(i,1) = 2 & \text{für } i \geq 1 \end{array}\right\} \text{Verankerung}$$

Für $i > 0, x > 1$: $A(i,x) = A(i-1, A(i, x-1))$

## Wertetabelle für $A(i,x)$

| $i$ \ $x$ | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 2 | 4 | 6 | 8 | 10 | 12 | 14 | ... |
| 1 | 0 | 2 | 4 | 8 | 16 | 32 | 64 | ... | |
| 2 | 0 | 2 | 4 | 16 | $2^{16}$ $=65536$ | $2^{65536}$ | | | |
| 3 | 0 | 2 | 4 | | | | | | |
| 4 | 0 | 2 | | | | | | | |

"Zeilenfunktion"

$2x = A(0,x)$

$2^x = A(1,x)$

$\underbrace{2^{2^{\cdot^{\cdot^{2}}}}}_{x} = A(2,x)$

$$A(1,2) = A(0, A(1,1)) = A(0,2) = 4$$

$$A(3,4) = 2^{2^{\cdots^{2}}}\Big\}\, 65536 \cdot x$$

## Inverse zu A

$$\alpha(m,n) = \min\{ i \mid A(i, \lfloor \tfrac{4m}{n} \rfloor) > \log n \}$$

Bsp.: $\alpha(24,8) = \min\{ i \mid A(i,3) > 3\} = 0$

$$\alpha(3 \cdot 2^{16}, 2^{16}) = \min(i \mid A(i,3) > 16\} = 3$$

Man sieht leicht, dass $\alpha(m,n) \leq 3$ für alle in der Praxis vorkommenden Werte für m & n ($\to$ Übung)

## Beweis des Satzes

Situation: n Knoten (Elemente)

$m > n$ Folge von $n-1$ Unions + m Finds

$U_1 F_1 F_2 U_2 U_3 F_3 \ldots . U F \ldots F_m U_{n-1}$

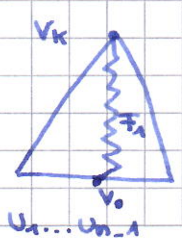
Am Ende: Einen Baum $T'$

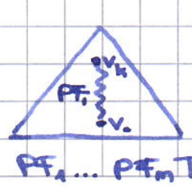
Können wir auch so erhalten: (konzeptionell)

a) Führe zunächst alle $n-1$ Unions aus $\to$ Baum T (log Tiefe)

b) Führe auf T m partielle Finds (partial finds) aus $PF_1, \ldots PF_m$

$PF_i$ simuliert das echte Find $F_i$, indem es den selben Pfad wie $F_i$ durchläuft (d.h. i.A. nicht bis zur Wurzel)




Wir schätzen nun die Kosten der Folge von Unions und partielles Finds ab.

Sei $\mathcal{F}$ die Multi-Menge (Menge in denen Elemente auch mehrfach vorkommen können) der Kanten, die in allen PFs durchlaufen werden.

zu zeigen: $|\mathcal{F}| = O(m \cdot \alpha(m,n))$

<u>Idee</u>

a) Teile $\mathcal{F}$ in Gruppen nach Rang der Endpunkt der Kanten ein (Rangdifferenz)

$$\text{Rang}(x) = \text{Höhe von } x \text{ im Baum } T \text{ (nicht } T'\text{)}$$

Rang(x) ist konstant (in Folge der PF's)

b) schätze alle Gruppen ab

Definitionen

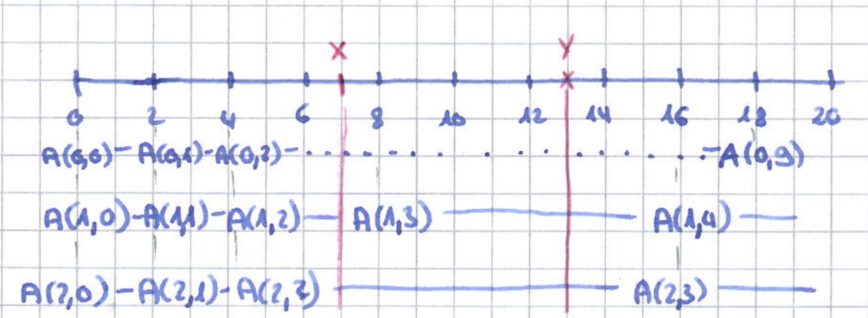
zunächst "Einteilung" der Knoten (nicht disjunkt)

Sei $z \in \mathbb{N}$ (wird später definiert $\to \alpha(m,n)$)

Für $0 \leq i \leq z$, $j \geq 0$ definiere

$$G_{ij} = \{\text{Knoten } x \mid A(i,j) \leq \text{Rang}(x) < A(i,j+1)\}$$

Veranschaulichen $\{[A(i,j) \dots A(i,j-1)]\}$ : Menge von Intervallen




Beispiel: $\text{Rang}(x) = 7$, $\text{Rang}(y) = 13$

$\Rightarrow x \in G_{0,3}, G_{1,2}, G_{2,2}, \dots$

$y \in G_{0,6}, G_{1,3}, G_{2,2}, \dots$

Einteilung der Kanten in $\mathcal{F}$

$$N_k = \{(x,y) \in \mathcal{F} \mid k = \min\{i \geq 0 \mid \exists j \text{ mit } x,y \in G_{ij}\}\}$$

für $0 \leq k \leq z$ im Bsp $(x,y) \in N_2$

und

$$N_{z+1} = \mathcal{F} \setminus \bigcup_{k=0}^{z} N_k \quad \text{(Rest)}$$

<u>Intuitiv</u>: $(x,y) \in N_i \Rightarrow$ Rangdiff$(x,y) \approx A(i,..)$ $\overset{?}{\Rightarrow}$ Übung)

<u>Im Beispiel</u>: Rang(x)=7, Rang(y)=13

$(x,y) \in N_2$ → erstes mal in selber Menge

$A(1,...)$ . . | . . .

$G_{2,2} = \underset{\overset{\shortparallel}{4}}{A(2,2)}$ . . | . $\underset{\overset{\shortparallel}{16}}{A(2,3)}$

Schließlich definiere:

$L_k := \{(x,y) \in N_k \mid (x,y)$ ist letzte (oberste) Kante auf PF-Pfad$\}$

für $0 \leq k \leq z+1$

<u>Lemma</u>

a) $|L_k| \leq m$ für $0 \leq k \leq z+1$

b) $|N_0 \setminus L_0| \leq n$

c) $|N_k \setminus L_k| \leq \frac{5}{8} n$ für $0 \leq k \leq z$

d) $|N_{z+1} \setminus L_{z+1}| \leq n \cdot a(z,n)$, wobei $a(z,n) = \min\{i \mid A(z,i) > \log n\}$

<u>Beweis</u>

zu a) Für jedes PF gibt es höchstens 1 Kante in $L_k$ (eine letzte Kante)
Da m PF's ⇒ Teil a

zu b) Sei $(x,y) \in N_0 / L_0$, dann gilt:

1) $\exists j$ (Spalte) mit $x, y \in G_{0,j}$

<u>dh</u> $\underset{\overset{\shortparallel}{2j}}{A(0,j)} \leq \text{Rang}(x) < \text{Rang}(y) < \underset{\overset{\shortparallel}{2(j+1)}}{A(0,j+1)}$ (Übung)

⇒ Rang(x) = 2j und Rang(y) = 2j+1
⇒ dh. Rangdifferenz = 1

2) $(x,y) \notin L_0$ ⇒ war nicht letzte Kante auf PF-Pfad

<u>PF-Pfad</u>:

PF(v)

es kann sein y=s


<u>Betrachte</u> PF (der (x,y) nach $N_0$ brachte

Dann $\exists (s,t) \neq (x,y)$ auf PF-Pfad mit $(s,t) \in N_0$
mit Rang(x) = 2j
Rang(y) = 2j+1
Rang(s) ≥ Rang(y)
Rang(t) > Rang(s) ⇒ $\overset{\text{Rang}(u)}{\text{Rang}(t)} \geq 2j+2$
= Rang(s)+1

Nach PF: x hat neuen Vater u (u=t ist möglich)
⇒ Rangdifferenz(x,u) ≥ 2
⇒ Spätere PFs können keine Kante (x,u) zu $N_0$ hinzufügen

$\Rightarrow$ Für alle Knoten $x$ gilt es höchstens eine Kante $(x,y) \in N_0 \setminus L_0$
$\Rightarrow |N_0 \setminus L_0| \leq n$

zu c)

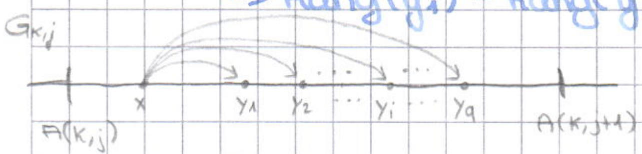
wie oben: Schätze Beitrag eines Knotens $x \in G_{k,j}$ zu $|N_k \setminus L_k|$ ab
dh. alle Kanten $(x, \dots)$ mit ...

Sei $k \geq 1$ und $x \in G_{k,j}$ beliebig (Spalte $j$)
dh $A(k,j) \leq Rang(x) < A(k,j+1)$

und $y_1, y_2, \dots, y_q$ alle Endknoten mit $(x, y_i) \in N_k \setminus L_k$ in der Reihenfolge in der sie von PFs benutzt werden

dh. $y_{i+1}$ ist Wurzel des PF von $(x, y_i)$

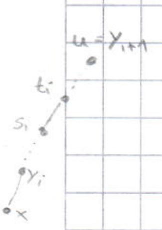
$\Rightarrow Rang(y_1) \leq Rang(y_2) \leq \dots \leq Rang(y_q) < A(k, j+1)$




Dann gilt:

i) $j \geq 2$ denn sonst wär $k = 0$ (min)

ii) $(x, y_i) \notin L_k$ für $i = 1, \dots, q$ $\Rightarrow \exists (s_i, t_i) \in N_k$ oberhalb von $(x, y_i)$ auf PF




$\Rightarrow Rang(x) < Rang(y_i) \leq Rang(s_i) < Rang(t_i) \leq Rang(y_{i+1})$

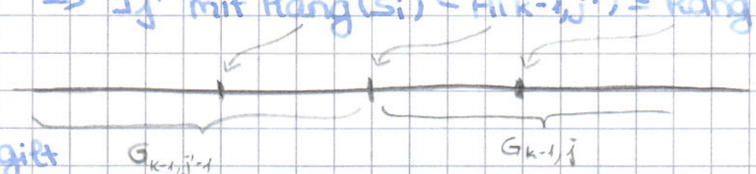
nach Pfadkomprimierung: $x$ hat Vater $y_{i+1}$

Definition von $N_k$ ($k$ minimal)

$\Rightarrow (x, y_i), (s_i, t_i) \in N_{k-1}$

$\Rightarrow \exists j'$ mit $Rang(s_i) < A(k-1, j') \leq Rang(t_i)$

Zeile n-1




und daher gilt

$Rang(y_i) < A(k-1, j) \leq Rang(y_{i+1})$

Iteriere: $q$-mal (für $y_1, \dots, y_q$)

$\Rightarrow \exists j''$ mit $Rang(y_i) < A(k-1, j'') \leq A(k-1, j''+q-1) \leq Rang(y_q)$

Umbenennung

I $\exists j$ mit $Rang(y_q) \geq A(k-1, j+q-1)$

Beobachtung: $j \geq 2$ (siehe oben)

Teile: $1 \leq k \leq z$

$x \in G_{n,j}$, $(x,y) \in N_r \Rightarrow y_1, \dots, y_q \in G_{k,j}$

$A(k,j) \le Rang(y_1) \le \dots \le Rang(y_q) < A(k,j+1)$

(II) $Rang(y_q) < A(k,j+1)$

(I+II) $A(k-1, j+q-1) < A(k,j+1) = A(k-1, A(k,j))$

$\overset{\text{Monotonie}}{\Rightarrow} \quad j+q-1 < A(k,j)$

$\overset{j \ge 2}{\Rightarrow} \quad q < A(k,j)$

<u>Das bedeutet</u> (Zusammenfassung)

Für jedes $x \in G_{k,j}$, $k \ge 1$, $j \ge 2$ gibt es höchstens $A(k,j)$ Kanten $(x,y) \in N_k \setminus L_k$

$$\Rightarrow |N_k \setminus L_k| \le \sum_{j \ge 2} |G_{k,j}| \cdot A(k,j)$$

* Behauptung: $|G_{k,j}| \le \frac{2n}{2^{A(k,j)}}$

Daraus folgt:

$$|N_k \setminus L_k| \le \sum_{j \ge 2} \frac{2n \cdot A(k,j)}{2^{A(k,j)}} \le 2n \cdot \sum_{j \ge 2} \frac{A(k,j)}{2^{A(k,j)}}$$

Es gilt: da $k \ge 1$, $A(k,j) \ge 2^j$

$$= 2n \sum_{j \ge 2} \frac{2^j}{2^{2^j}} = 2n \sum_{j \ge 2} \frac{1}{2^{2^j - j}} = 2n \cdot \Big( \overset{j=2}{\frac{1}{4}} + \underbrace{\overset{3}{\frac{1}{32}} + \overset{4}{\frac{1}{2^{12}}} + \dots}_{\le 1/16} \Big)$$

$$\le 2n \cdot \frac{5}{16} = \frac{5}{8} n$$

* Beweis der Behauptung: $|G_{k,j}| \le \frac{2n}{2^{A(k,j)}}$ mit $k \ge 1$, $j \ge 2$

$G_{k,j}$ = Knoten mit Rang im Intervall $[A(k,j) \dots A(k,j+1)[$

Sei $\ell$ eine bel. natürliche Zahl mit $A(k,j) \le \ell < A(k,j+1)$

Wir zählen alle $x \in G_{k,j}$ mit $Rang(x) = \ell$

Sei $G_{k,j,\ell} = \{ x \in G_{k,j} \mid Rang(x) = \ell \}$

Es gilt:

i) Jeder Knoten $x$ mit Rang $\ell$ hat $\ge 2^\ell$ Nachkommen (Gewicht von $x \ge 2^\ell$) ← weighted union

ii) Für $x, y$ mit $Rang(x) = Rang(y)$ und $x \ne y$ sind die Nachfolgermengen disjunkt

$$\text{i)+ii)} \Rightarrow |G_{k,j,\ell}| \le \sum_{\ell=A(k,j)}^{A(k,j+1)-1} |G_{k,j,\ell}| \le \sum_{\ell=A(k,j)}^{\infty} \frac{n}{2^\ell} = n \cdot \sum_{\ell=A(k,j)}^{\infty} \frac{1}{2^\ell}$$

$$= \frac{1}{2^{A(k,j)}} + \frac{1}{2} \cdot \frac{1}{2^{A(k,j)}} + \frac{1}{4} \cdot \frac{1}{2^{A(k,j)}} = \frac{n}{2^{A(k,j)}} \cdot \left(1 + \frac{1}{2} + \frac{1}{4} + \dots\right)$$

$$\leq \frac{2n}{2^{A(k,j)}}$$

zu d) $k = z+1$ auch hier Schranke $q$

weighted union $\Rightarrow$ $Rang(y_q) \leq \log n$ (da T' Höhe $\leq \log n$)

I) für $k = z+1$

$\exists j: Rang(y_q) \geq A(k-1, j+q-1)$ mit $j \geq 2$

$\log n \geq Rang(y_q) \geq A(z, j+q-1)$

$\Rightarrow A(z, j+q-1) \leq \log n$

$\Rightarrow j+q-1 < a(z,n)$, da $a(z,n)$ minimal mit $A(z, a(z,n)) > \log n$

$\overset{j\geq 2}{\Rightarrow} q \leq a(z,n)$

Also gibt es höchstens $a(z,n)$ Kanten $(x,y)$ für jeden Knoten $x$ in $N_{z+1} \setminus L_{z+1}$

$\Rightarrow |N_{z+1} \setminus L_{z+1}| \leq n \cdot a(z,n)$ ■ a,b,c,d

Zurück zu Satz (Tarjan)

Kosten aller $m \geq n$ Finds

$$|F| = \underbrace{\sum_{k=0}^{z+1} |L_k|}_{a} + \underbrace{|N_0 \setminus L_0|}_{b} + \underbrace{\sum_{k=1}^{z} |N_k \setminus L_k|}_{c} + \underbrace{|N_{z+1} \setminus L_{z+1}|}_{d}$$

$$|F| \leq \underbrace{(z+2) \cdot m + n + z \cdot \frac{5}{8} n}_{A} + \underbrace{n \cdot a(z,n)}_{B}$$

Das gilt für jedes $z$!

Betrachte $z = \alpha(m,n)$, $m \geq n$

i) $A = O(m \cdot \alpha(m,n))$

ii) $a(z,n) = a(\alpha(m,n), n) \overset{\text{Def von } a}{=} \min\{j \mid A(\alpha(m,n), j) > \log n\}$

mit $\alpha(m,n) = \min\{i \mid A(i, \lfloor \frac{4m}{n} \rfloor) > \log n\}$

$= \leq \lfloor \frac{4m}{n} \rfloor$

$\Rightarrow B \leq n \cdot \lfloor \frac{4m}{n} \rfloor = O(m)$

$A + B = O(m \cdot \alpha(m,n))$

Satz: $n-1$ Unions und $m \geq n$ Finds kosten $\Theta(m \cdot \alpha(m,n))$

i) nicht linear $\longleftrightarrow$ praktisch: $O(m)$, da $\alpha(m,n) \leq 3$ für realistische Werte

ii) praktisch sehr effizient (kleine Konstante)

iii) leicht implementierbar, Korektheit einfach

iv) obere Schranke ist scharf d.h. es gibt Probleminstanzen mit dieser Laufzeit

v) die Komplexität des Problems ist tatsächlich

$\Omega(m \cdot \alpha(m,n))$ untere Schrank

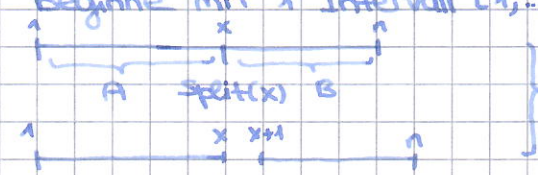
Union / Find $\longleftrightarrow$ split / Find (→ Übung)

$\downarrow$

Mengen sortiert, Splitten an einer Stelle mit $\leq, >$

$\Rightarrow$ Menge von Intervallen (disjunkt) von $x$ in Teile $A \leq x$ und $B > x$

beginne mit 1 Intervall $[1, \dots, n]$




max $n-1$ Splits

Finds wie vorher