Das UNION-FIND-Problem

Algorithmus von Kruskal:

- Zu Beginn startet man mit n Einzelkomponenten.
- Liegen u und v in verschiedenen Komponenten K_1 und K_2 ?
- Wenn ja, werden K_1 und K_2 vereinigt: $K = K_1 \cup K_2$

UNION-FIND (Verwaltung disjunkter Teilmengen)

- Grundmenge G von n Elementen
- Verwaltung einer Zerlegung von G in disjunkte Teilmengen (Partition)
- Zu Beginn ist jedes Element eine Einzelmenge $\{v_1\}, \{v_2\}, \ldots\ldots, \{v_n\}$
- M = FIND(v) bestimmt die „Menge", zu der v gehört.
- UNION(M_1, M_2) ersetzt die Mengen M_1 und M_2 durch $M_1 \cup M_2$

M ist „abstrakt" (Es wird nicht spezifiziert, was M ist.)

- FIND(u) = FIND(v) $\Leftrightarrow$ u und v gehören zur gleichen Menge
- FIND(u), FIND(v) kann als Argument für UNION verwendet werden.

M kann z.B. ein ausgewähltes Element (ein Repräsentant) der Menge sein.
oder ein anderes Objekt, das die Menge repräsentiert
oder eine Nummer

Algorithmus von Kruskal:

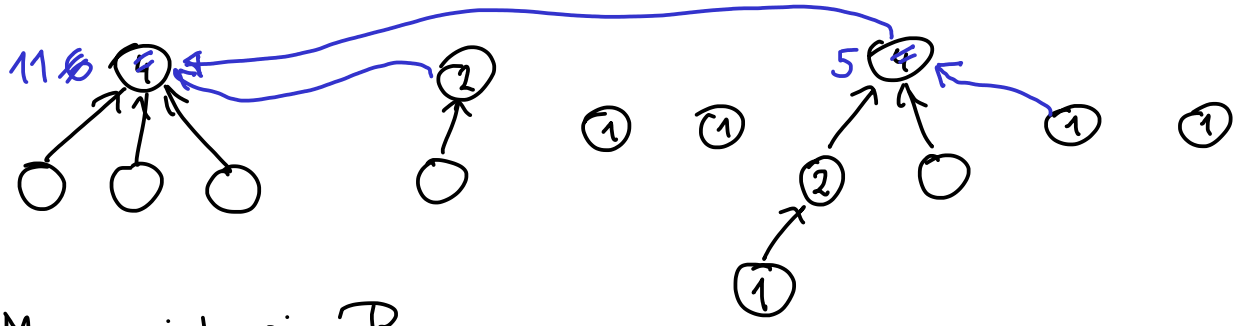
- Zu Beginn startet man mit n Einzelkomponenten. ✓
- Liegen u und v in verschiedenen Komponenten K_1 und K_2 ?

  $K_1 := \text{FIND}(u)$; $K_2 := \text{FIND}(v)$
- Wenn ja, werden K_1 und K_2 vereinigt: $K = K_1 \cup K_2$

  if $K_1 \neq K_2$: UNION (K_1, K_2) ; $T := T + uv$

$2m \times$ FIND. $(n-1) \times$ UNION

Darstellung der Mengen als Bäume
Vereinigung nach Größe

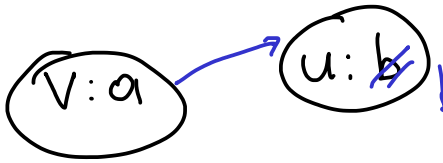


Jede Menge ist ein Baum.
Die Wurzel ist der Repräsentant der Menge;
sie speichert zusätzlich die Größe der Menge.
Jede Nichtwurzel hat einen Zeiger zum Elternknoten.

FIND: Folge den Zeigern bis zur Wurzel.

UNION: Hänge die Wurzel der kleineren Baumes als Kind an die Wurzel des größeren Baumes.

Behauptung: Der Teilbaum des Elternknotens von v ist mindestens doppelt so groß wie der Teilbaum von v.



$b \geq a$

$b + a \geq 2a$ ✓

Die Höhe der Bäume ist höchstens $\log_2 n$.

FIND: $O(\log n)$

UNION: $O(1)$

⇒ Algorithmus von Kruskal:

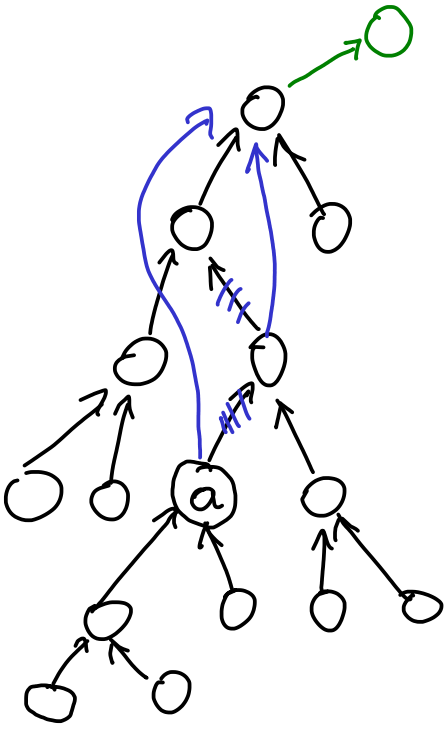
$O(m \log n)$ ... Sortieren der Kanten

$O(m \log n)$ ... FIND

$(m \geq n-1)$ $O(n)$ ... UNION

$O(m \log n)$

Verbesserung durch Pfadkompression, inverse Ackermann-Funktion α(n)



Robert Tarjan $(m+n) \cdot \alpha(n)$

- $a+1$
- $a+b$
- $a \cdot b$
- a^b
- $2^{2^{2^{\cdot^{\cdot^{\cdot^{2}}}}}}$ (a twos) $2 \uparrow a$
- ...
-

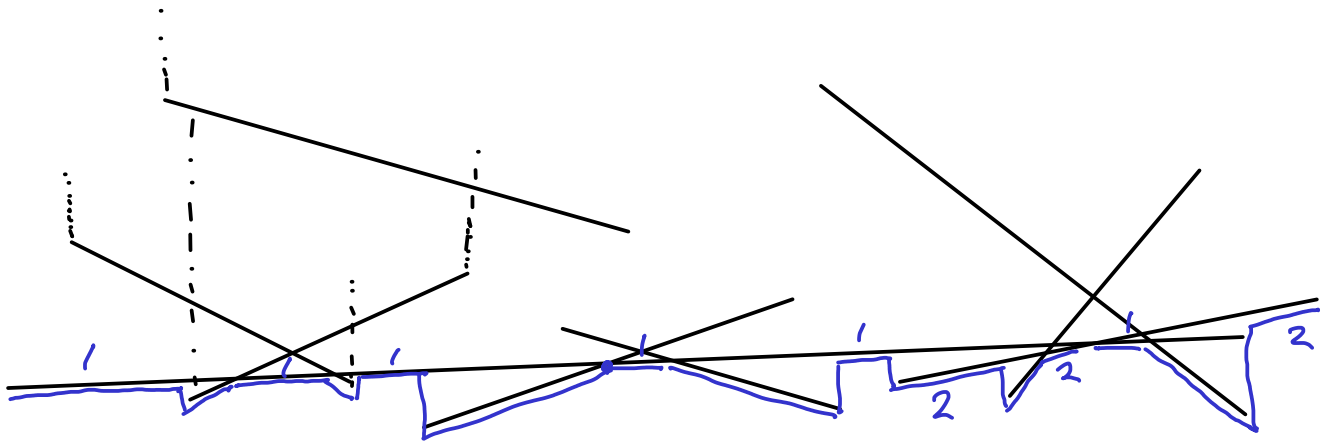
-
- $-$
- $/2$
- $\log n = \min\{i \mid \underbrace{n/2/2/2 \cdots /2}_{i} \le 1\}$
- $\log^* n = \min\{i \mid$

  $\vdots$

  $\underbrace{\log \log \log \log \cdots \log}_{i} n \le 1\}$

- f
- $f^*(n) = \min\{i \mid \underbrace{f(f(f(\cdots f}_{i}(n))) \le 1\}$

$$\alpha(n) = \min\{k \mid \overbrace{(((((\log^*)^*)^*)\cdots)^*}^{k}(n) \le 2\}$$



$\Theta(n\,\alpha(n))$