3.1 Union Find jest $O(\log^* n)$

- Analizujemy wersję UnionFinda z kompresją ścieżek oraz podwieszaniem mniejszego drzewa (wierzchołkowo) pod większe.
- Przez większość czasu będziemy analizować złożoność tak jakby kompresji ścieżki nie było.
- W każdej chwili, jeśli jakieś drzewo (a nawet poddrzewo) ma wysokość h to ma ono $\geqslant 2^h$ wierzchołków. (dowód bardzo przyjemny przez indukcje)
- Na odwrót: jeśli jakieś poddrzewo ma conajmniej n wierzchołków to jego wysokość jest $\leqslant \lceil \log n \rceil$.
- Już w tej chwili otrzymujemy złożoność m instrukcji na Union Findzie $O(n+m \lg n)$, bo findy wykonują nie więcej niż $\lg n$ operacji, a uniony dwie operacje find.
- Rangą wierzchołka x nazywamy wysokość poddrzewa zaczepionego w wierzchołku x (jakie ono by było gdyby nie było kompresji ścieżek; mimo że były!)
- Ile jest wierzchołków o randze r? Ponieważ każdy wierzchołek o randze r ma conajmniej 2^r potomków, może być ich co najwyżej $\frac{n}{2^r}$.
- Jeśli wierzchołek x jest właściwym (to znaczy nie jest y przyp. tłumacz) potomkiem y, to $rank(x) < rank(y)$. (dowód - kompresja ścieżek nie zmienia tego)
- Definiujemy funkcje exp z gwiazdkami:

$$\exp^*(n) = \begin{cases} 1 & \text{jeśli } n = 0 \\ 2^{\exp^*(n-1)} & \text{wpp} \end{cases}$$

- i funkcję log z gwiazdką:

$$\log^*(n) = \min_i \{\exp^*(i) \geqslant n\}$$

- Wprost z definicji wynika, że:

$$\log^*(r) = g \Leftrightarrow \exp^*(g-1) < r \leqslant \exp^*(g)$$

- Definiujemy grupę wierzchołka jako $\log^*$ z jego rangi:

$$group(x) = \log^*(rank(x))$$

- Wierzchołków, w grupie g jest:

$$N(g) \leqslant \sum_{r=\exp^*(g-1)+1}^{\exp^*(g)} \frac{n}{2^r} \leqslant \frac{n}{2^{\exp^*(g-1)+1}}[1 + \frac{1}{2} + \frac{1}{4} + \frac{1}{8} + ...] = \frac{n}{\exp^*(g)}$$

- Okej - chcemy teraz lepiej oszacować ile płacimy za operacje Find.
- Koszt operacji Find na wierzchołku x jest proporcjonalny do ścieżki jaką x musi pokonać w drodze do korzenia.
- Załóżmy, że ścieżka ta składa się z wierzchołków $x_1, x_2, ..., x_n$ gdzie $x = x_1$, a x_n to korzeń.
- Za przejście z wierzchołka x_i do wierzchołka x_{i+1} płacimy w dwóch walutach:
- Jeśli $group(x_i) \neq group(x_{i+1})$ albo gdy $x_{i+1} = x_n$ (dochodzimy do korzenia) płacimy w dolarach.
- Jeśli $group(x_i) \neq group(x_{i+1})$ to wierzchołek x_i płaci w euro.
- Dowolny wierzchołek ma rangę co najwyżej $\lceil \log n \rceil$, zatem różnych grup jest nie więcej niż $\log^* \lceil \log n \rceil$.
- Zatem zapłacę co najwyżej $m \log^* \lceil \log n \rceil \in O(m \log^* n)$ dolarów.
- Po każdej operacji Find, wierzchołek x_i jest podczepiony do wierzchołka o niższej randze niż uprzednio był.
- Zatem wierzchołek, który jest w grupie g płaci co najwyżej tyle euro, ile jest różnych rang w grupie g. Jest ich dużo, bo $\exp^*(g) - exp^*(g-1) - 1 \in O(\exp^*(g))$.

- W grupie g jest $n/\exp^*(g)$ wierzchołków, więc sumarycznie wszystkie one zapłacą $n/\exp^*(g) \times \exp^*(g) = n$ euro.
- Ponieważ różnych grup jest $O(\log^* n)$, wszystkie wierzchołki nie zapłacą więcej niż $O(n \log^* n)$.
- Sumujemy dolary i euro i kończymy dowód.