Základní myšlenky

Jako úplně první si stanovíme, že budeme veškeré dělení hromad zaokrouhlovat nahoru. Například u čísla 35, při dělení do dvou hromad bychom doopravdy získali hromady s výškou 17 a 18, ale pro veškeré naše účely můžeme říct, že máme dvakrát 18. Toto si můžeme dovolit právě proto, že rozdělení hromady s výškou 18 nám zabere jedno operace a stále nám zbyde hromada s výškou 17, tedy celkově provedeme 18 instrukcí. Kdybychom chtěli rozdělit obě hromady, tak získáme 8, 9, 9, 9, a narážíme opět na stejnou situace, kde je nám vlastně jedno, že jedna hromada je nižší, protože máme další, vyšší hromady. Tímto si můžeme dovolit veškeré dělení zaokrouhlovat nahoru. A to, že vůbec dělíme hromady na hromady o stejné velikosti bereme jako samozřejmost.

V předchozím odstavci jsme si vlastně také ukázali, že když máme hromady o výšce n a $n + 1$, tak nemá smysl dělit jenom jednu z těch hromad. Toto můžeme dále rozšířit na případ, kdy máme x krát výšku n, tak je výhodné rozdělit tyto hromady právě tehdy, kdy nejvyšší hromada menší než n má výšku menší (nebo rovno) $n - x$. Ve spojení s předchozím nálezem, můžeme rovnou seskupit všechny množiny právě x čísel, které se nachází v intervalu $n \pm x$.

Jako další si ukážeme, že výsledný počet operací je $\Omega(\sqrt{N})$, $\mathcal{O}(N)$ (počítáme s tím, že alespoň jedna hromada má maximální možnou výšku). Toto si jednoduše vysvětlíme pomocí dvou extrémních případů.

V prvním případě budeme mít $N - 1$ hromad o výšce 1 a poté jednu hromadu o výšce N. Jediné co nás zajímá je dělení poslední hromady, a je zřejmé, že nejvýhodnější je dělení jeho odmocninou, tedy $\sqrt{N}$. Je nám jedno, jestli tuto odmocninu zaokrouhlíme nahoru či dolu, v obou případech se nám bude výsledek lišit pouze o jedna. Případné další zvětšení dělení nám také zlepší výsledek pouze o 1, jestli vůbec. Pro snažší pochopení si můžeme představit číselnou osu, kde náš dělitel začne vlevo u jedničky a výsledek vpravo u N. Můžeme sledovat, že posunutím dělitele blíž k $\sqrt{N}$ se ná musí o dost víc blížit výsledek k $\sqrt{N}$, protože musí putovat mnohem větší vzdálenost. Přímo v $\sqrt{N}$ se setkají (protože $\sqrt{N}^2 = N$ pro kladný N), a dalším posunutím dělitele doprava se obrátí sitauce, tedy to už není výhodný.

V druhém případě budeme mít N hromad o výšce N. Je zřejmé, že aby jakékoliv dělení bylo výhodné, je potřeba ho provést na všechny hromady, tedy N krát. Ale pomocí N operací jsme rovnou schopni všechny hromady redukovat na nulu. V zadání je zmíněno, že hromada má výšku *řádově* N, ne přímo N, ale na asymptotickou složitost výsledku to nic nemění.

Algoritmus

Nejdřív si uvědomíme, že algoritmus nikdy nepřekoná $\mathcal{O}(N)$, protože už jenom procházení a určení nejvyšší hromady trvá $\mathcal{O}(N)$. Zároveň také existují vstupy, které nelze řešit rychleji než $\mathcal{O}(N)$ operací. S touto myšlenkou jsme schopni "bezpečně" setřídit naše hromádky pomocí přehrádkové třízení a zachovat tím lineárnost řešení. Zbytek algoritmu není příliš složitý: začneme od největší hromady a zkontrolujeme, jestli je výhodný zvětšit jeho dělitel o jeden. Jestli ano, tak to provedeme, ale jestli ne, tak to znamená, že zvětšení dělitele této hromady je výhodný pouze tehdy, kdy zmenšíme dělitel i další hromady. Toto nám umožní ty dvě hromady seskupit (zůstane nejvyšší z nich, ale zvětší se počet potřebných operací pro zvětšení dělitele) a pokračovat dál.

Proces kontroly, jestli je výhodný zvětšit dělitele nějaké hromady, funguje na bázi porovnání s maximem, ale naše hromádky už nebudou setřízené už od první dělení hromady. Jak tedy budeme pamatovat jaké hromady mohou být nejvyšší? Po prvním dělení máme 2 možnosti, druhou nejvyšší hromadu nebo nově vzniklou hromadu, a rovnou si je oba hodíme do max—heapu. Už je jasné, že si všechny maxima budeme ukládat do haldy, ale abychom zachovali lineárnost řešení, tak je potřeba ukázat, že naše halda nebude mít velikost $\mathcal{O}(N)$.

Začneme myšlenkou, že když máme několik čísel se stejným dělitelem, tak největší z nich je maximem z té skupiny, tedy budeme pamatovat pouze jeho ze všech čísel se stejným dělitelem. Jinými slovy, v haldě bude maximálně tolik čísel, jaký je největší dělitel, a největší dělitel jsme už stanovili jako $\sqrt{N}$. Z toho už je jasné omezení velikosti haldy a vložení a smazání nastane ve stejném okamžiku, tedy $\sqrt{N}$ krát. Tedy veškeré naše operace s haldou se nám vejde do $\mathcal{O}(\sqrt{N} \cdot \log(\sqrt{N}))$.

V rámci práci s haldou provedeme veškeré minimalizování nejvyšších hromady a zbyde nám jenom process házení kyseliny, dokud nám nezmizí všechny hromady. Tento krok můžeme vlastně provést v konstatním čase, ale počet operací ve vstuput nám nakonec vyjde (jak jsme stanovili na začátku) někde v intervalu $\langle\sqrt{N}; N\rangle$.

Celková časová komplexita češení je tedy $\mathcal{O}(N)$ a prostorová komplexita (když nepočítáme vstup a výstup) je $\mathcal{O}(\sqrt{N})$.