Řešení úlohy č. 4
Inverze

Nejprve vyřešíme úlohu pro případ $N = K$. Rozmysleme si, jaké počty inverzí dokážeme vytvořit. Nula inverzí dostaneme, pokud všechna čísla poskládáme vzestupně, tzn. $1, 2, \dots, N$. Jednu inverzi lze získat, pokud prohodíme N a $N-1$, čímž dostaneme $1, \dots, N-2, N, N-1$. Další inverzi získáme, pokud N opět prohodíme s předchozím číslem, tentokrát $N-2$. Postupným prohazováním získáme posloupnost $N, 1, 2, \dots, N-1$, která má $N-1$ inverzí. Tento postup můžeme analogicky opakovat s číslem na konci posloupnosti a to $N-1$. Nejvíce inverzí získáme, pokud vytvoříme posloupnost $N, N-1, \dots, 1$. Každé 2 prvky budou tvořit inverzi, jejich počet je $\binom{N}{2}$. Pokud bude $N < K$, pak stačí zvolit čísla $1, \dots, N$ a situace je stejná.

Nyní vyřešme zbývající případ, kdy $N > K$. Bez újmy na obecnosti řekněme, že zvolíme do posloupnosti čísla $1, \dots, K$. Označme frekvenci čísla x v posloupnosti jako f_x. Předpokládejme, že máme posloupnost s nejvíce inverzemi. Nyní si (sporem) dokážeme, že pro všechny dvojice $x, y : x \neq y$ platí, že $|f_x - f_y| \leq 1$. Jistě platí, že tato posloupnost je sestupně seřazená. Pokud by v ní bylo nějaké číslo na pozici i, které je větší než jeho předchůdce na pozici $i-1$, tak prohodíme čísla na těchto pozicích a získali jsme další inverzi, což je spor s předpokladem, že máme posloupnost s maximálním možným počtem inverzí. Pro spor vyberme dvě čísla x, y taková, že $f_x - f_y > 1$. Nyní změňme jedno z x na y. V prvé řadě ukažme, že se změní počty inverzí pouze mezi čísly x a y a počty inverzí mezi jinými čísly a x nebo y nebudou ovlivněny. Součet inverzí mezi x a všemi čísly většími než $\max(x, y)$ a y a všemi čísly většími než $\max(x, y)$ se z definice určitě nezmění. Mezi všemi čísly většími než $\max(x, y)$ a x a y se z definice počty inverzí nezmění. Obdobně to bude fungovat čísly menšími než $\min(x, y)$. Pro všechna čísla l mezi $\max(x, y)$ a $\min(x, y)$ platí, že pokud před změnou byla inverze mezi x a l (tzn. $x > l > y$), pak po změně bude táž inverze mezi l a y. Stejný argument bude platit, pokud $y > x$. Tímto jsme ukázali, že změna jednoho x na y nijak neovlivní počty inverzí se zbývajícími čísly. Víme, že mezi všemi x a všemi y je $f_x \cdot f_y$ inverzí (každé x s každým y). Tento vztah nyní přepišme na $f_x \cdot (f_x - k)$, kde $k := f_x - f_y$ a víme, že $k \geq 2$. Chceme ukázat, že změnou jednoho x na y dostaneme více než $f_x(f_x - k)$ inverzí. Začněme tedy provádět úpravy.

$$f_x(f_x - k) \leq (f_x - 1)(f_x - k + 1)$$
$$f_x^2 - f_x k \leq f_x^2 - f_x k + k - 1$$
$$0 \leq k - 1$$

Dostali jsme se k tomu, že $k - 1 \geq 0$, což je pravda vždy. To nám však říká, že jsme změnou jednoho x na y získali více inverzí, což je spor s předpokladem, že naše posloupnost již měla největší možný počet inverzí. **Tl; dr** optimální řešení v sobě má všechna čísla stejně často, příp. se frekvence 2 skupin čísel mohou lišit o jedničku.

V předchozí sekci jsme si naznačili algoritmus, který běží v $O(N^2)$, a také jsme si určili, jaká čísla se ve výsledné posloupnosti budou nacházet. Zbývá nám přijít na to, jak prvky poskládat za sebe s přesně R inverzemi v lineárním čase. Jeden ze způsobů si nyní ukážeme.

Prvně vyrobíme vzestupnou posloupnost A čísel $1, 1, \dots, 1, 2, \dots, 2, \dots, K$ tak, aby frekvence odpovídaly frekvenci obrácené posloupnosti z předchozích odstavců. Dále nastavíme počítadlo inverzí C na 0 a připravíme si pole F, kde na i-tém indexu nastavíme frekvenci čísla i ve výsledné posloupnosti. Každý prvek pole F nastavíme na $\lfloor N/K \rfloor$ a zbývajících

$N - \lfloor N/K \rfloor K$ prvků přerozdělíme mezi prvky pole postupně od začátku. Budeme potřebovat i pole P, ve kterém budeme udržovat prefixový součet pole F, abychom pro každý člen a_i dokázali rychle zjistit, kolik v poli existuje menších čísel. Nyní jsme připraveni vytvořit výslednou posloupnost opakováním následujícího postupu. Podíváme se na poslední prvek posloupnosti A, který označíme a. Zjistíme, co to je za číslo a podíváme se v poli P na prvek P_{a-1}, což značí, kolik existuje čísel menších než a. To nám řekne, kolik získáme inverzí, když člen a posuneme na začátek posloupnosti A. Pokud $C + P_{a-1} \leq R$, pak a přesuneme na začátek posloupnosti, k počítadlu C přičteme P_{a-1} a postup opakujeme. V opačném případě víme, že kdybychom číslo A přesunuli, pak získáme více inverzí, než se chce v zadání. Proto budeme číslo A postupně prohazovat se sousedem před ním a pokaždé zvýšíme C o jedna. Algoritmus končí ve chvíli, kdy $C = R$. Pokud tímto postupem získáme sestupnou posloupnost a $C < R$, pak řešení neexistuje.

Rychlejší algoritmus nezískáme, protože minimálně musíme posloupnost vypsat, což zabere $O(N)$ času. Paměťová složitost je taktéž lineární vzhledem k délce výsledné posloupnosti.

Poznamenejme, že toto není jediné správné řešení.