

3. Korman je gurmán

Bruno Petrus – Gymnázium Bajkalská

Algoritmus, ktorý používa náš program, funguje následovne. Nájdeme 2 najväčšie čísla, pozrieme sa či sú v správnom poradí resp. či väčšie čísla je v rade ďalej. Ak nie presuň menšie číslo na začiatok. Následne opakuj s menšími číslami.

Tento algoritmus funguje, lebo v usporiadanom rade je vždy väčšie za menším. Náš algoritmus len pozerá, kde toto neplatí v pôvodnom rade a na základe toho urobí úpravy. Nájde najmenší počet krokov, lebo vzhľadom na pravidlá úlohy o úpravách pola, prvok môžeme posunúť len na začiatok. Vďaka tomu že ideme od najväčšieho čísla a nie napríklad od indexu 0 docielime, že vždy budú 2 čísla vo vzájomne správnom poradí na najmenší počet krokov.

Nás program využíva dátovú štruktúru `std::multimap`. Multimap je utriedená key-value dátová štruktúra. Dôvod pre výber multimap bol, že je utriedená od najväčšieho po najmenšie hodnoty kľúča a toto nám umožnilo rýchle iterovanie cez pole v zostupnom poradí prvkov. Taktiež má veľmi rýchlu časovú zložitosť pri access-e, čo potrebujeme, keď meníme value daného key. V ostatných testovacích sadách môže pole obsahovať viacero tých istých čísel, preto sme si vybrali multimap a nie map.

Program začne tým, že načíta vstup do multimap, kde key je hodnota a value poradie. Vzhľadom na to že multimap je utriedená od najmenšieho po najväčšie, musíme iterovať cez ňu v opačnom poradí. Vždy sa pozrieme na väčšie číslo a menšie číslo, zistíme či sú v poradí alebo nie. Ak nie tak musíme dať menšie z nich na začiatok. Musíme si dať pozor na pole, kde sa nachádza viackrát ten istý prvok, musíme preto upraviť všetky takéto prvky.

Pamäťová zložitosť je rovná n , musíme si pamätať všetky prvky. Časová zložitosť je v najhoršom prípade, keď je proste celé pole naopak ako má byť, rovné $O(n)$.