

## Relaxovaný AVL strom

Nedávný úspěšný návrh intervalového binárního vyhledávacího stromu povzbudil profesora Fabinarise Suchbauma v jeho dalším výzkumném úsilí. Nyní se se svým týmem plánuje zaměřit na evaluaci relaxovaných verzí AVL stromu.

Relaxovaný AVL strom povoluje pro každý uzel, aby absolutní hodnota rozdílu výšek levého a pravého podstromu byla až  $D$ , kde  $D$  je zvolená konstanta větší než 1. Vyvažování takového stromu se provádí pomocí rotací L, R, LR a RL stejně jako v případě standardního AVL. Pro AVL i relaxovaný AVL strom uvažujeme následující pravidla:

- Kdykoliv nevyvážený uzel může být vyvážen více typy rotací, potom se jednoduchá rotace upřednostňuje před dvojitou rotací.
- Jestliže klíč  $key$  je již ve stromě uložen, potom  $Insert(key)$  neprovede žádnou změnu.
- Jestliže klíč  $key$  není ve stromě uložen, potom  $Delete(key)$  neprovede žádnou změnu.
- Pokud při  $Delete(key)$  je klíč  $key$  uložen v uzlu  $U$ , který má dva potomky, potom se náhradní klíč pro  $U$  hledá v levém podstromě uzlu  $U$ .

## Úloha

Proveďte danou posloupnost operací  $Insert$  a  $Delete$  nad iniciálně prázdným AVL stromem a relaxovaným AVL stromem. Pro oba stromy vypište konečnou hloubku a počet rotací vykonaných během celého procesu.

## Vstup

Vstup sestává z jednoho řádku, který obsahuje pět kladných celých čísel  $D, A, B, M, N$  oddělených mezerami.  $D$  je parametr relaxovaného AVL stromu udávající maximální povolený rozdíl mezi výškami levého a pravého podstromu pro každý uzel. Čísla  $A, B, M, N$  definují posloupnost klíčů  $S = (s_1, s_2, \dots, s_N)$  následovně:

$$s_1 = B,$$

$$s_{k+1} = (A \times s_k + B) \bmod M, k = 1, 2, \dots, N-1.$$

Operace  $Insert(s_1), Insert(s_2), \dots, Insert(s_N)$  jsou provedeny v tomto pořadí na prázdném AVL a relaxovaném AVL stromu. Následovány jsou operacemi  $Delete(s_3), Delete(s_6), Delete(s_9), \dots, Delete(s_{N-(N \bmod 3)})$  v tomto pořadí (každý třetí klíč vygenerované posloupnosti je smazán).

Platí:  $D \leq 10, 1 \leq A, B, M \leq 10^9, 1 \leq N \leq 10^6$ .

## Výstup

Výstup sestává ze dvou řádků. První řádek obsahuje čísla  $H$  a  $R$  oddělená mezerou.  $H$  je hloubka AVL stromu po provedení všech operací,  $R$  je sumární počet jednoduchých a dvojitych rotací vykonaných v AVL stromu během vyvažování (každá dvojitá rotace se do hodnoty  $R$  započítává pouze jednou, nepovažuje se za dvě jednoduché rotace). Druhý řádek má stejný tvar a obsahuje hloubku a počet rotací pro relaxovaný AVL strom.

## Příklad 1

### Vstup

2 1 4 11 5

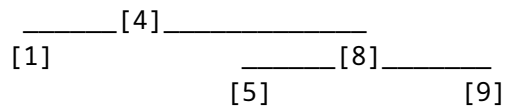
### Výstup

2 1  
2 0

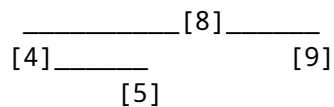
Následující schémata zachycují stavy stromů při provádění operací v Příkladu 1.

AVL strom:

Insert(4), Insert(8), Insert(1), Insert(5), Insert(9)

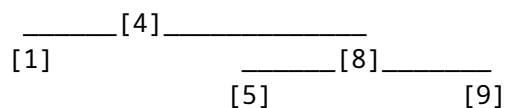


Delete(1) - provede L rotaci v uzlu [4].

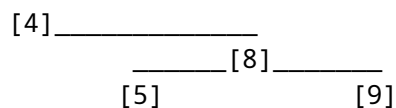


relaxovaný AVL strom (D=2):

Insert(4), Insert(8), Insert(1), Insert(5), Insert(9)



Delete(1)



## Příklad 2

### Vstup

2 9 9 16 8

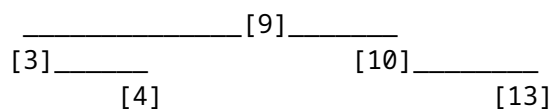
### Výstup

2 3  
3 1

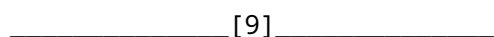
Následující schémata zachycují stavy stromů při provádění operací v Příkladu 2.

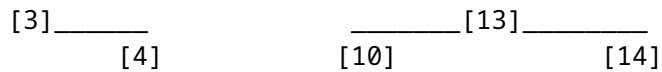
AVL strom:

Insert(9), Insert(10), Insert(3), Insert(4), Insert(13)

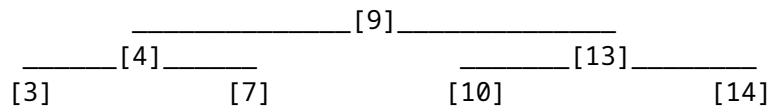


Insert(14) - provede L rotaci v uzlu [10]

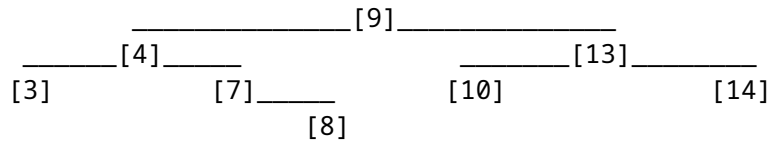




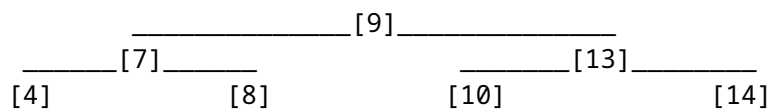
Insert(7) - provede L rotaci v uzlu [3]



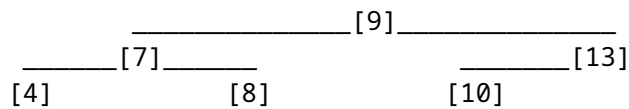
Insert(8)



Delete(3) - provede L rotaci v uzlu [4]

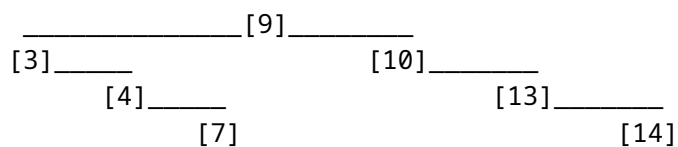


Delete(14)

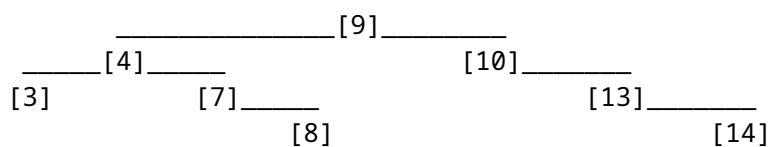


relaxovaný AVL strom (D=2):

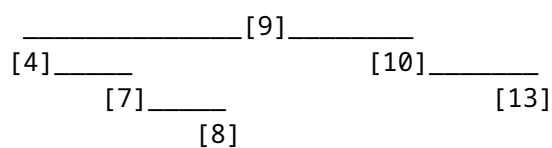
Insert(9), Insert(10), Insert(3), Insert(4), Insert(13), Insert(14), Insert(7)



Insert(8) - provede L rotaci v uzlu [3]



Delete(3), Delete(14)



## Veřejná data

Veřejná data k úloze jsou k dispozici. Veřejná data jsou uložena také v odevzdávacím systému a při každém odevzdání/spuštění úlohy dostává řešitel kompletní výstup na stdout a stderr ze svého programu pro každý soubor veřejných dat.

[Veřejná data](#)