

## Intervalové operace v BVS

Profesor Fabinaris Suchbaum z Institutu Maxe Plancka v Saarbrückenu je uznávaným odborníkem na binární vyhledávací stromy (BVS) a jejich vylepšení. Nedávno byl osloven společností Data4Ever, aby analyzoval chování BVS za dvou předpokladů:

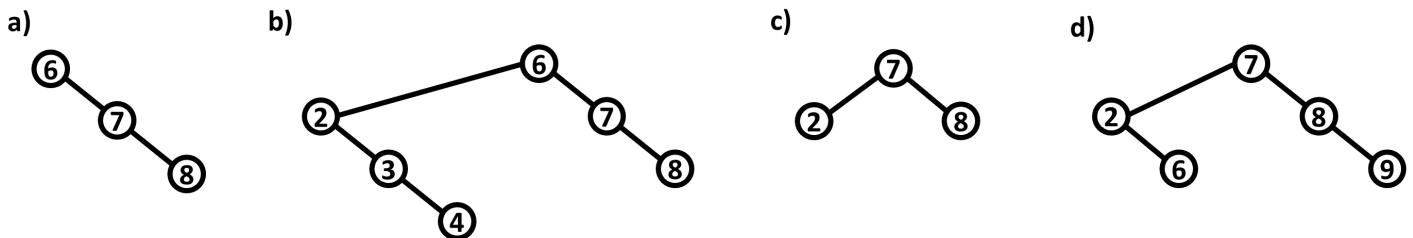
1. Do BVS se často pomocí operace Insert vkládají posloupnosti po sobě jdoucích klíčů.
2. Často se také mažou sekvence klíčů spadající do určitého intervalu.

Pro účely testování popsaného scénáře navrhl profesor Suchbaum pro BVS intervalové operace  $\text{Insert}(a, b)$  a  $\text{Delete}(a, b)$  s následujícími vlastnostmi:

1. Výsledek operace  $\text{Insert}(a, b)$  aplikované na BVS  $T$  odpovídá sekvenčnímu vložení klíčů  $a, a+1, \dots, b$  do  $T$  v uvedeném pořadí. Pokud  $T$  již nějaký klíč z daného intervalu obsahuje, není tento klíč znovu vložen.
2. Výsledek operace  $\text{Delete}(a, b)$  aplikované na BVS  $T$  odpovídá smazání klíčů  $a, a+1, \dots, b$  v  $T$ , avšak v pořadí, které splňuje, že při každé dílčí operaci  $\text{Delete}(\text{key})$  podstrom uzlu s klíčem  $\text{key}$  neobsahuje kromě  $\text{key}$  žádné jiné hodnoty z intervalu  $[a, b]$ . Operace  $\text{Delete}(\text{key})$  je definovaná tak, že při mazání uzlu  $u$  se dvěma potomky se hledá klíč, který v  $u$  nahradí klíč  $\text{key}$ , v podstromu zakořeněném v pravém potomku uzlu  $u$ . V případě, že strom  $T$  neobsahuje klíč  $\text{key}$ , neprovede  $\text{Delete}(\text{key})$  v  $T$  žádnou změnu.

### Úloha

Pro prázdný BVS proveďte zadanou posloupnost intervalových operací Insert a Delete.



**Obrázek 1.** Stavby na začátku prázdného BVS po provedení operací: a)  $\text{Insert}(6, 8)$ , b)  $\text{Insert}(2, 4)$ , c)  $\text{Delete}(3, 6)$ , d)  $\text{Insert}(6, 9)$ . Výsledek operace  $\text{Delete}(3, 6)$  aplikovaný na BVS b) odpovídá podle definice postupnému smazání klíčů v pořadí 4, 3, 6 (uzel s klíčem 3 nelze smazat jako první, protože podstrom zakořeněný v tomto uzlu obsahuje klíč 4, který je také určený ke smazání).

### Vstup

První řádek vstupu obsahuje celé číslo  $N$ , které udává počet provedených intervalových operací. Následuje  $N$  řádků. Na každém z těchto řádků jsou hodnoty  $O, A, B$  oddělené mezerami.  $A$  a  $B$  jsou celá nezáporná čísla.  $O$  je buď znak 'i', potom řádek reprezentuje operaci  $\text{Insert}(A, B)$ , a nebo znak 'd', potom řádek reprezentuje operaci  $\text{Delete}(A, B)$ .

Platí  $1 \leq N \leq 5 \times 10^5$  a  $0 \leq A \leq B \leq 7 \times 10^4$ .

### Výstup

Výstup obsahuje jeden textový řádek, na kterém jsou dvě čísla oddělená mezerou. První číslo je rovno počtu uzlů výsledného stromu po provedení všech intervalových operací a druhé číslo je rovno hloubce tohoto stromu.

### Příklad 1

#### Vstup

```
4
i 6 8
i 2 4
d 3 6
i 6 9
```

#### Výstup

```
5 2
```

Data a řešení Příkladu 1 můžeme vidět na **Obrázku 1**.

### Příklad 2

#### Vstup

```
6
i 3 3
i 5 10
d 1 3
d 8 9
i 15 20
d 10 16
```

**Výstup**

```
7 6
```

---

**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](#)