

#### פתרון שאלה 4 בממ"ן 17

נשתמש בעץ אדום-שחור מורחב (עץ ערכי מיקום)  $T$ .  
בנוסף לשדה  $size$ , יהיה בכל צומת בעץ שדה  $sum$ , שבו נשמור את סכום המפתחות של האיברים הנמצאים בתת-עץ המושרש בצומת.  
כמו כן נחזיק מצביע  $p2$  לאיבר השני הקטן ביותר בעץ.

נתאר את אופן ביצוע הפעולות השונות:

INSERT( $S, k$ ): הכנסה רגילה לעץ ערכי מיקום;  
עדכון השדה  $sum$  מתבצע בדומה לעדכון השדה  $size$ ;  
עדכון המצביע  $p2$ : אם  $k < key[p2]$  אז  $p2 \leftarrow \text{PREDECESSOR}(p2)$   
זמן הריצה הכולל:  $O(\lg n)$ .

DELETE( $S, z$ ): מחיקה רגילה מעץ ערכי מיקום;  
עדכון השדה  $sum$  מתבצע בדומה לעדכון השדה  $size$ ;  
עדכון המצביע  $p2$ : אם  $key[z] \leq key[p2]$  אז מחשבים מחדש את ערך המצביע:  
 $p2 \leftarrow \text{SUCCESSOR}(\text{MINIMUM}(\text{root}[T]))$   
זמן הריצה הכולל:  $O(\lg n)$ .

MEDIAN( $S, k_1, k_2$ ): מחפשים ב- $S$  את  $k_1$  ו- $k_2$  ובעזרת השגרה OS-RANK מוצאים את המיקום  $r_1$  של  $k_1$  ואת המיקום  $r_2$  של  $k_2$ ; בעזרת השגרה OS-SELECT מוצאים את הערך שדירוגו  $\lfloor (r_1 + r_2) / 2 \rfloor$ ;  
בפסידו-קוד:

```
MEDIAN( $S, k_1, k_2$ )
 $x_1 \leftarrow \text{SEARCH}(T, k_1)$ 
 $x_2 \leftarrow \text{SEARCH}(T, k_2)$ 
 $r_1 \leftarrow \text{OS-RANK}(T, x_1)$ 
 $r_2 \leftarrow \text{OS-RANK}(T, x_2)$ 
 $r \leftarrow \lfloor (r_1 + r_2) / 2 \rfloor$ 
return OS-SELECT( $\text{root}[T], r$ )
```

זמן הריצה:  $O(\lg n)$ .

הערה: מניחים שהמפתחות  $k_1$  ו- $k_2$  נמצאים ב- $S$ .

$SUM(S, k)$  : מתחילים מהשורש ומתקדמים במורד העץ. אם  $k$  גדול שווה מערך המפתח בצומת הנוכחי, אז מעדכנים את סכום המפתחות שערכם אינו עולה על  $k$  ופונים ימינה; אחרת פונים שמאלה;  
בפסידו-קוד :

```

SUM(S, k)
x ← root[T]
total_sum ← 0
while x ≠ NIL
    do if k ≥ key[x]
        then total_sum ← total_sum + sum[left[x]] + key[x]
           x ← right[x]
        else x ← left[x]
return total_sum

```

זמן הריצה :  $O(\lg n)$ .

$MIN2(S)$  : מחזירים את  $key[p2]$  ;

זמן הריצה :  $O(1)$ .