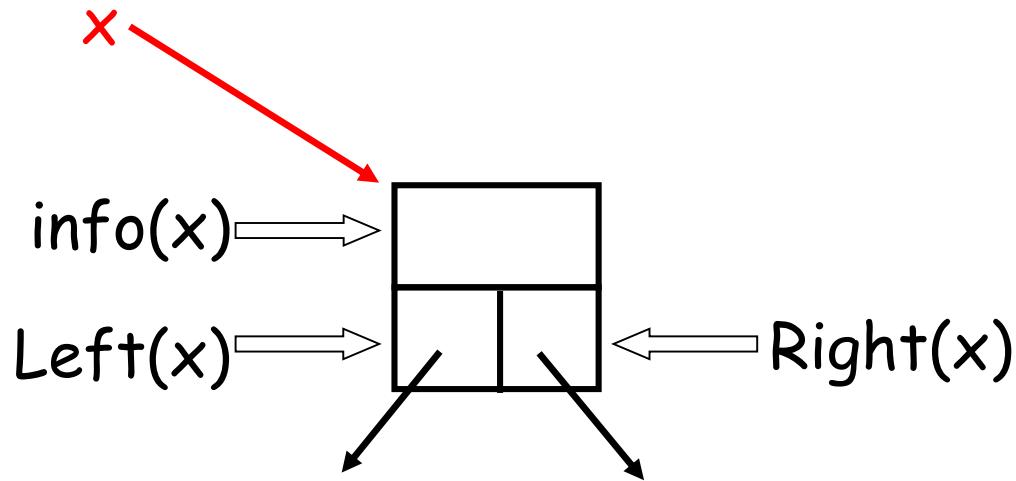


# יצוג עצים ביבאים

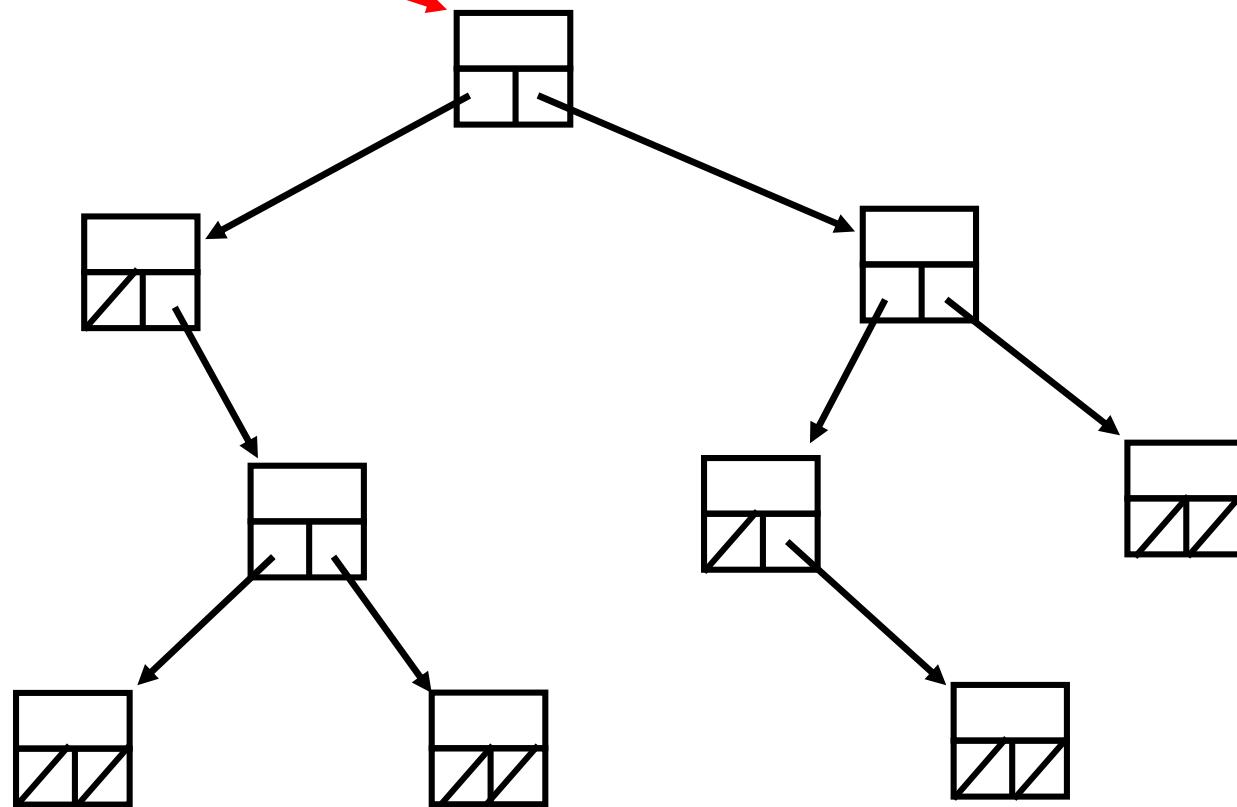
יצוג של צומת:



# יצוג עצים ביבאריים

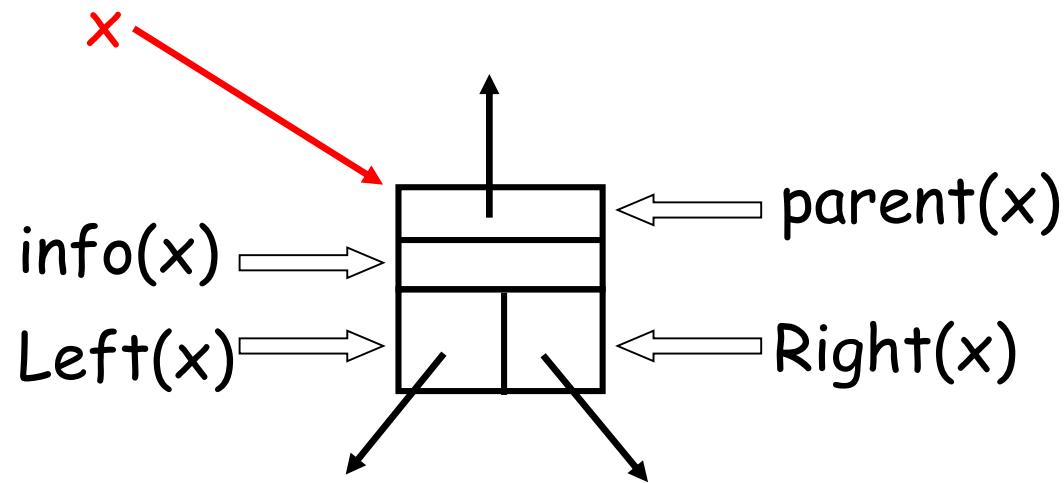
Root( $T$ )

יצוג של עץ:



# יצוג עצים ביבאים

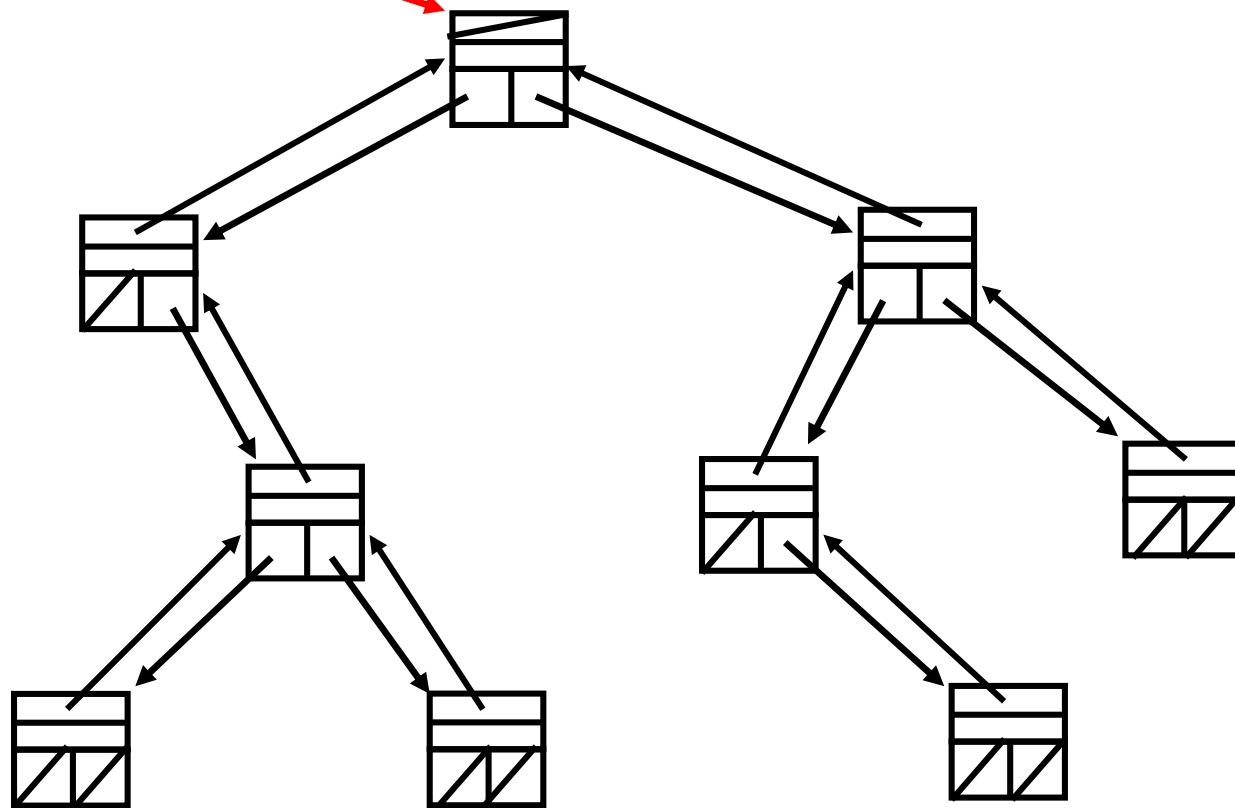
יצוג של צומת:



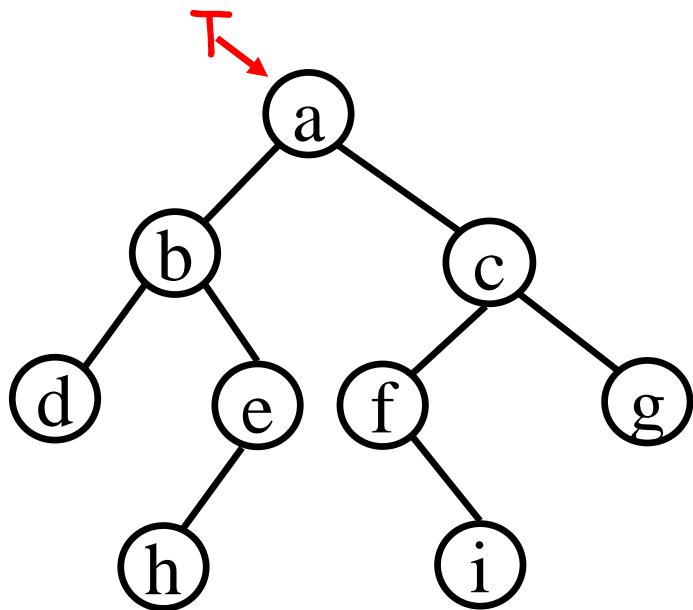
# יצוג עצים ביבאים

Root( $T$ )

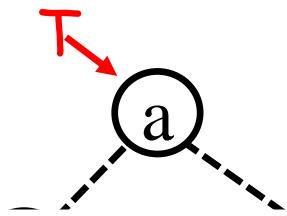
יצוג של עץ:



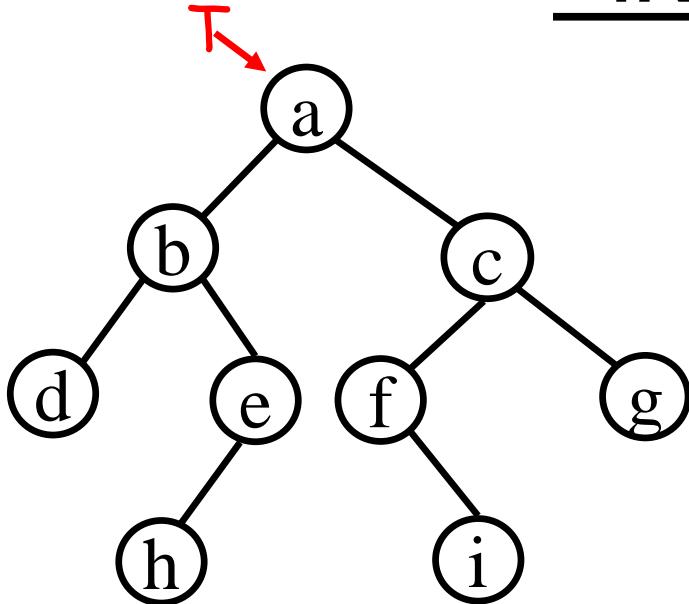
# סרייה עץ בינארי



# סרייה עז בינארי



# סרייה עץ בינארי



**סרייה תומכת inorder**

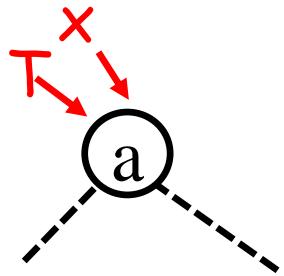
סrox את תת העץ השמאלי

בקר בשורש

סrox את תת העץ הימני

הדפסת כל מפתחות העץ בסדר inorder

# סרייה עץ בינארי



**סרייה תואנית inorder**

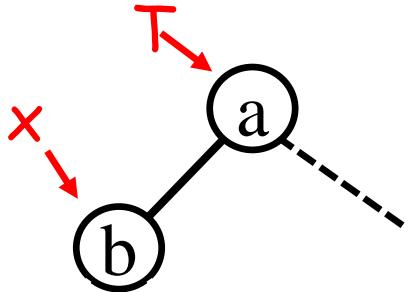
סrox את תת העץ השמאלי

בקר בשורש

סrox את תת העץ הימני

הדפסת כל מפתחות העץ בסדר inorder

# סרייה עץ בינארי



**סרייה תואנית inorder**

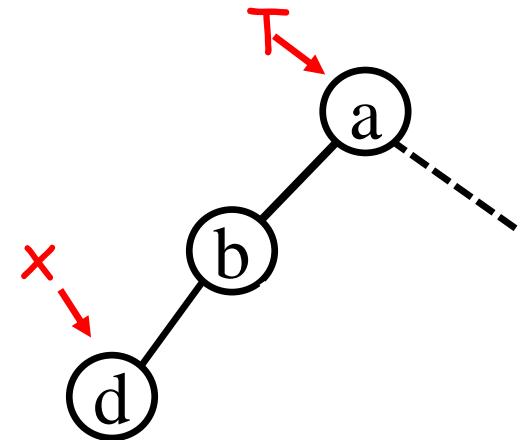
סrox את תת העץ השמאלי

בקר בשורש

סrox את תת העץ הימני

הדפסת כל מפתחות העץ בסדר inorder

# סרייה עץ בינארי



**סרייה תומכת בעץ בינארי**

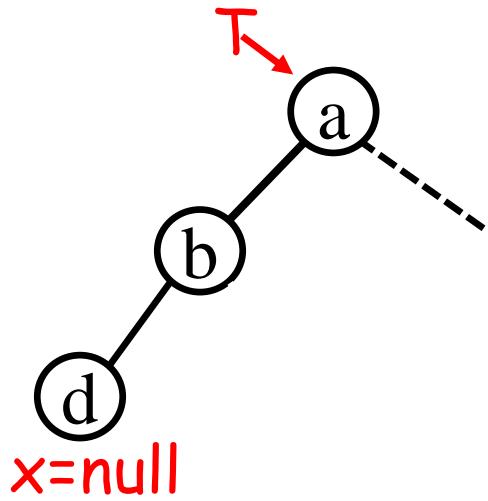
סrox את תת העץ השמאלי

בקר בשורש

סrox את תת העץ הימני

הדפסת כל מפתחות העץ בסדר inorder

# סרייה עץ בינארי



**סרייה תואנית inorder**

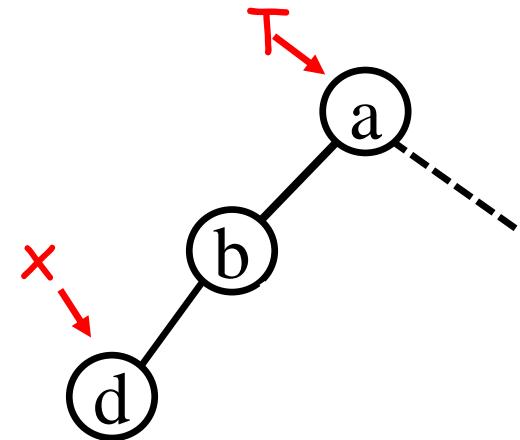
סrox את תת העץ השמאלי

בקר בשורש

סrox את תת העץ הימני

הדפסת כל מפתחות העץ בסדר inorder

# סרייה עץ בינארי



**סרייה תומכת inorder**

סrox את תת העץ השמאלי

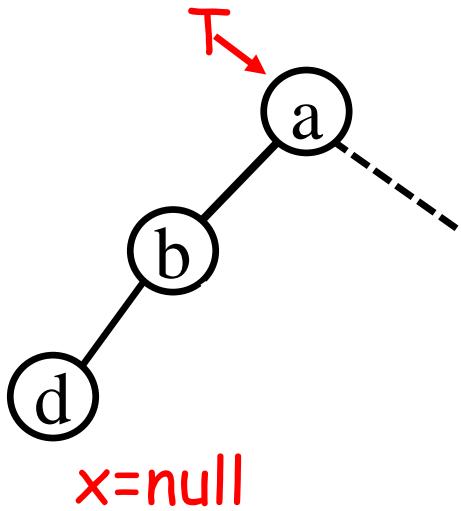
**בקר בשורש**

סrox את תת העץ הימני

הdfs כל מפתחות העץ בסדר inorder

d

# סרייה עץ בינארי



d

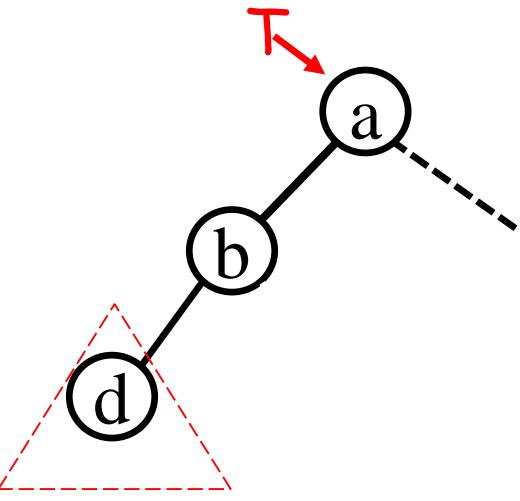
## סרייה תומכת **inorder**

סrox את תת העץ השמאלי  
בקר בשורש

סrox את תת העץ הימני

הדפסת כל מפתחות העץ בסדר **inorder**

# סרייה עץ בינארי



**סרייה תומכת *inorder***

סrox את תת העץ השמאלי

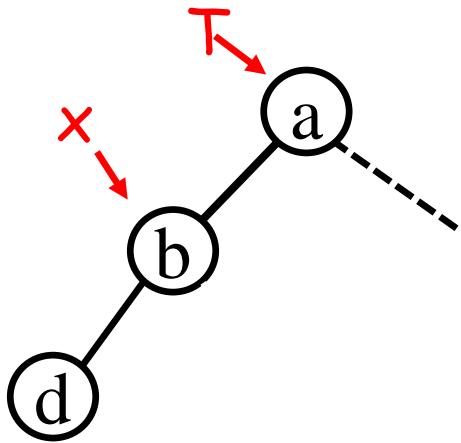
בקר בשורש

סrox את תת העץ הימני

הdfsת כל מפתחות העץ בסדר *inorder*

d

# סרייה עץ בינארי



**סרייה תומכת inorder**

סrox את תת העץ השמאלי

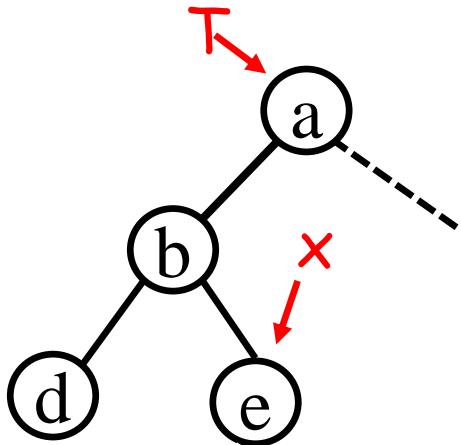
**בקר בשורש**

סrox את תת העץ הימני

הdfsת כל מפתחות העץ בסדר inorder

d b

# סרייה עץ בינארי



**סרייה תומכת inorder**

סrox את תת העץ השמאלי

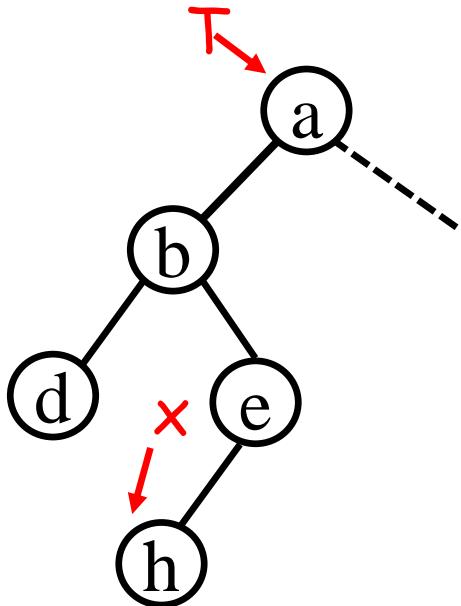
בקיר בשורש

סrox את תת העץ הימני

הdfs כל מפתחות העץ בסדר inorder

d b

# סרייה עץ בינארי



d b

**סרייה תואנית inorder**

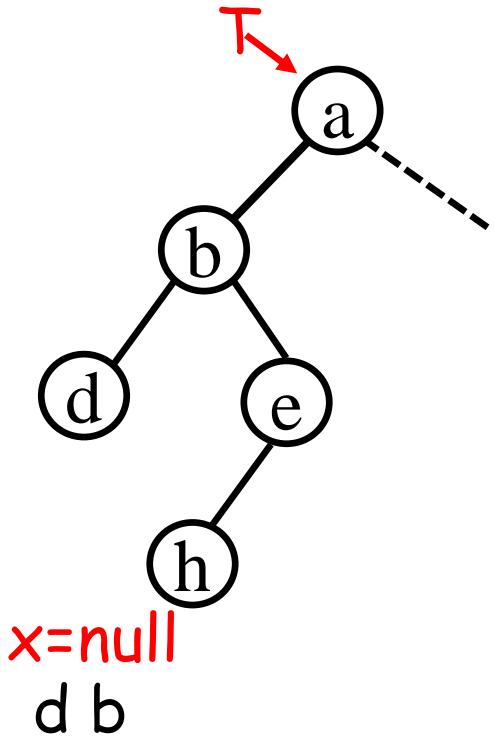
סrox את תת העץ השמאלי

בקר בשורש

סrox את תת העץ הימני

הdfs כל מפתחות העץ בסדר inorder

# סרייה עץ בינארי



**סרייה תואנית inorder**

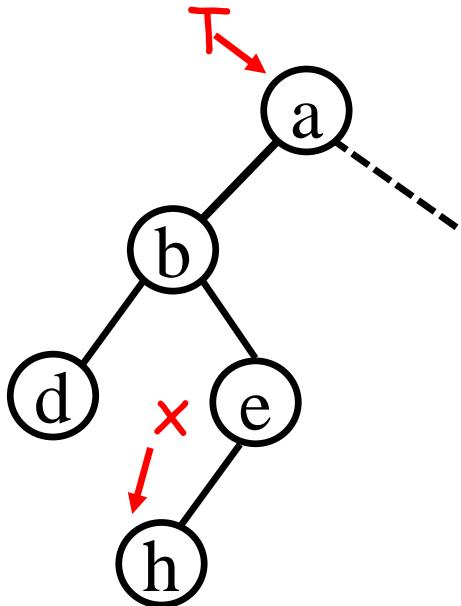
סrox את תת העץ השמאלי

בקר בשורש

סrox את תת העץ הימני

הדפסת כל מפתחות העץ בסדר inorder

# סרייה עץ בינארי



d b h

**סרייה תומכת inorder**

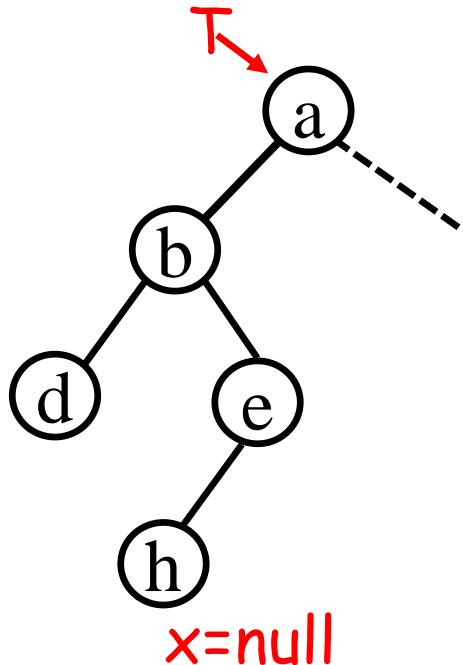
סrox את תת העץ השמאלי

**בקר בשורש**

סrox את תת העץ הימני

הdfs כל מפתחות העץ בסדר inorder

# סרייה עץ בינארי



**סרייה תואנית inorder**

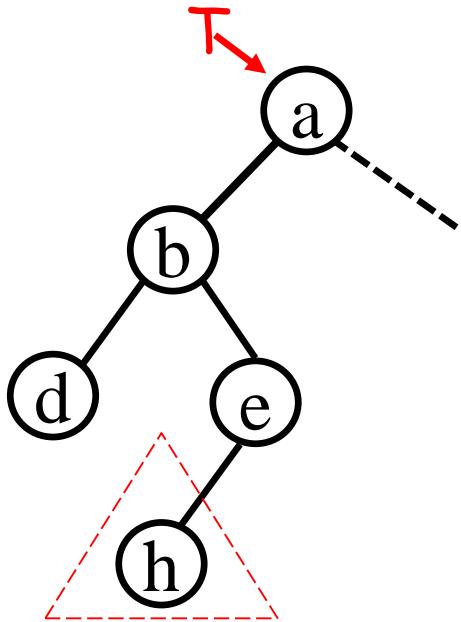
סrox את תת העץ השמאלי  
בקר בשורש

סrox את תת העץ הימני

הדפסת כל מפתחות העץ בסדר inorder

d b h

# סרייה עץ בינארי



d b h

## סרייה תומכת **inorder**

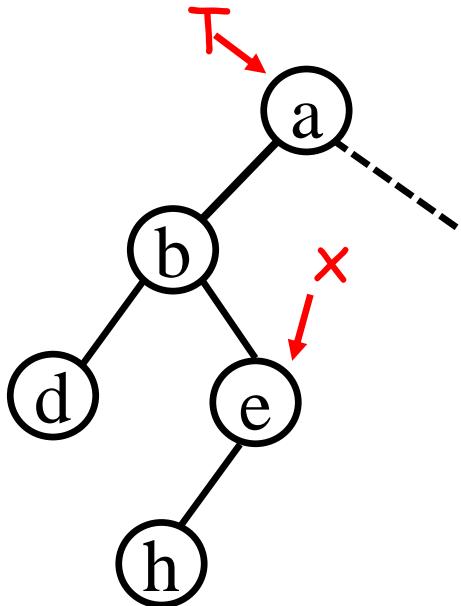
סrox את תת העץ השמאלי

בקר בשורש

סrox את תת העץ הימני

הdfsת כל מפתחות העץ בסדר **inorder**

# סרייה עץ בינארי



d b h e

**סרייה תומכת inorder**

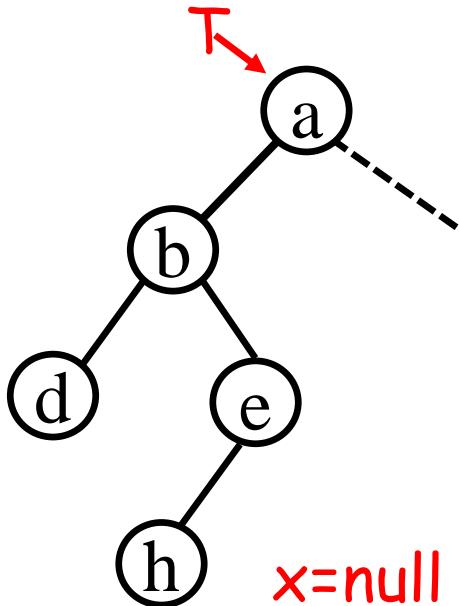
סrox את תת העץ השמאלי

**בקר בשורש**

סrox את תת העץ הימני

הdfsת כל מפתחות העץ בסדר inorder

# סרייה עץ בינארי



d b h e

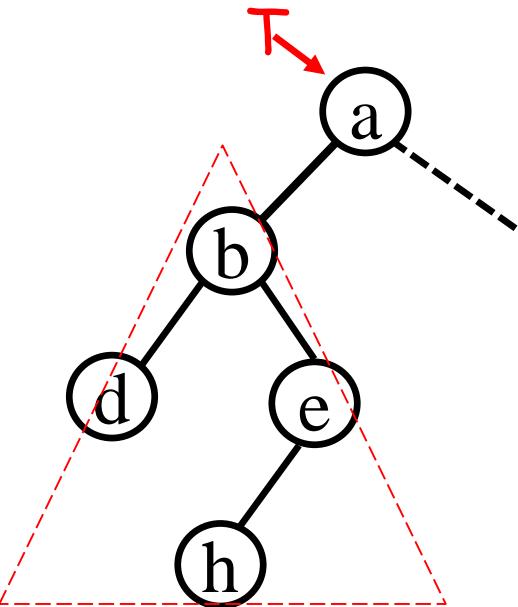
**סרייה תומכת inorder**

סrox את תת העץ השמאלי  
בקיר בשורש

סrox את תת העץ הימני

הדפסת כל מפתחות העץ בסדר inorder

# סרייה עץ בינארי



d b h e

## סרייה תומכת **inorder**

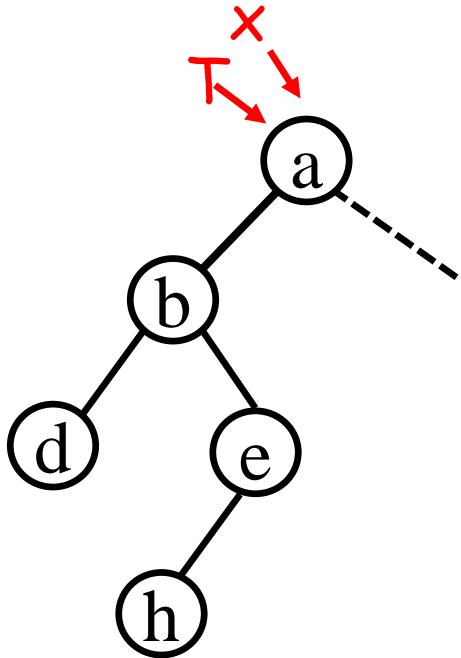
סrox את תת העץ השמאלי

בקיר בשורש

סrox את תת העץ הימני

הדפסת כל מפתחות העץ בסדר **inorder**

# סרייה עץ בינארי



d b h e a

**סרייה תומcit inorder**

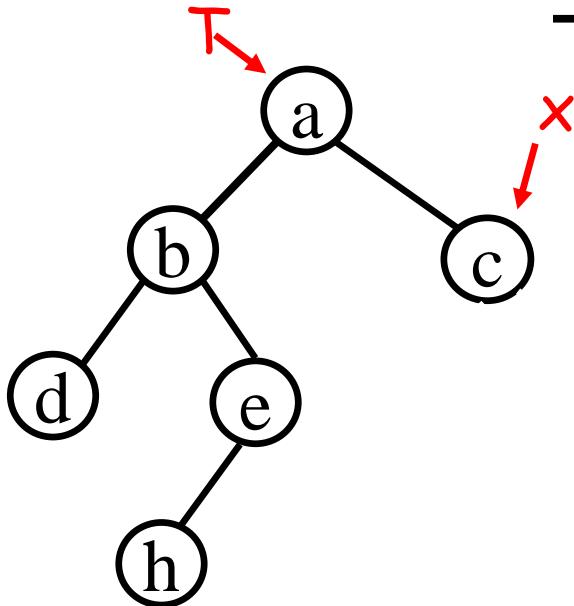
סrox את תת העץ השמאלי

**בקר בשורש**

סrox את תת העץ הימני

הdfsת כל מפתחות העץ בסדר inorder

# סרייה עץ בינארי



d b h e a

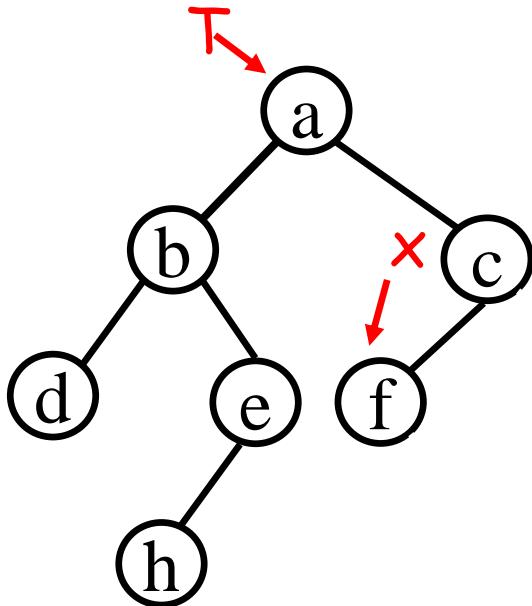
**סרייה תומכת inorder**

סרווק את תת העץ השמאלי  
בקיר בשורש

סרווק את תת העץ ימני

הdfs את כל מפתחות העץ בסדר inorder

# סרייה עץ בינארי



d b h e a

**סרייה תואנית inorder**

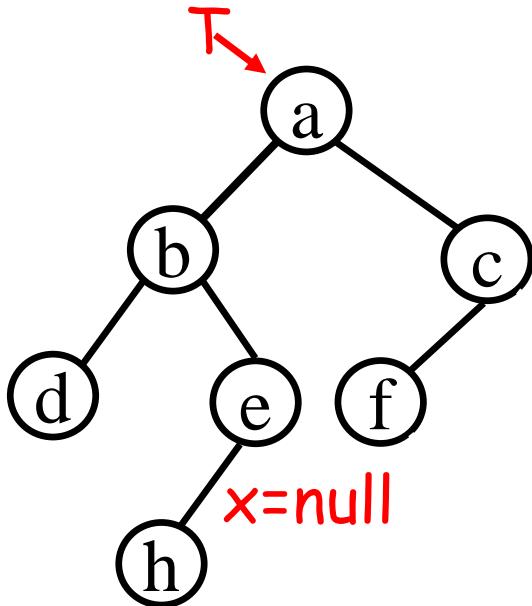
סrox את תת העץ השמאלי

בקר בשורש

סrox את תת העץ הימני

הdfs כל מפתחות העץ בסדר inorder

# סרייה עץ בינארי



**סרייה תואנית inorder**

סrox את תת העץ השמאלי

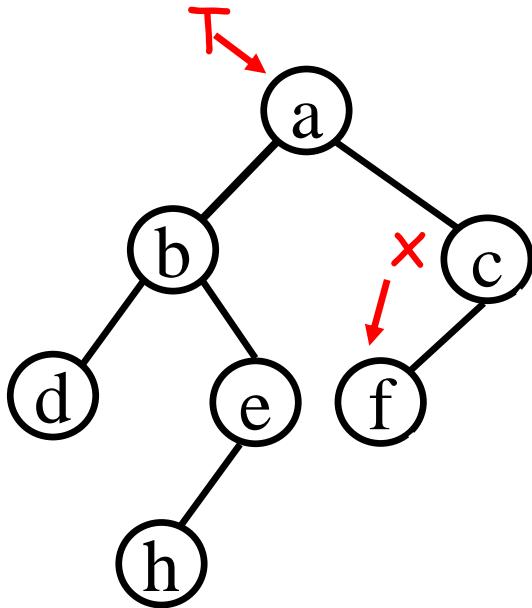
בקר בשורש

סrox את תת העץ הימני

הדפסת כל מפתחות העץ בסדר inorder

d b h e a

# סרייה עץ בינארי



**סרייה תומכת inorder**

סrox את תת העץ השמאלי

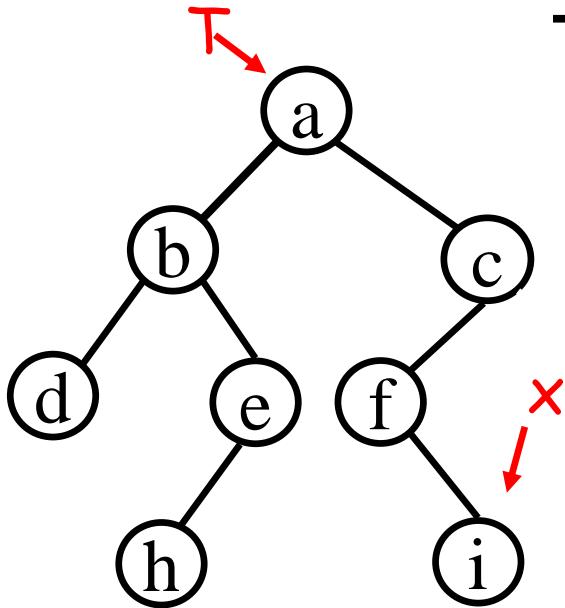
**בקר בשורש**

סrox את תת העץ הימני

הדפסת כל מפתחות העץ בסדר inorder

d b h e a f

# סרייה עץ בינארי



d b h e a f

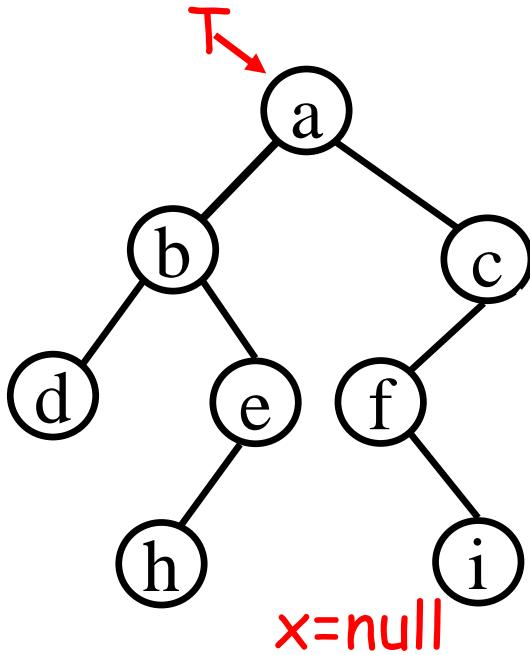
**סרייה תומכת inorder**

סrox את תת העץ השמאלי  
בקר בשורש

**סrox את תת העץ ימני**

הdfsת כל מפתחות העץ בסדר inorder

# סרייה עץ בינארי



d b h e a f

**סרייה תואנית inorder**

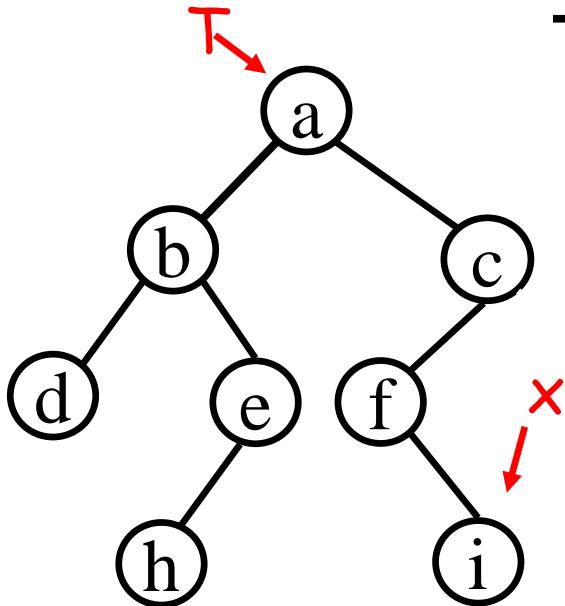
סrox את תת העץ השמאלי

בקר בשורש

סrox את תת העץ הימני

הדפסת כל מפתחות העץ בסדר inorder

# סרייה עץ בינארי



d b h e a f i

**סרייה תומכית inorder**

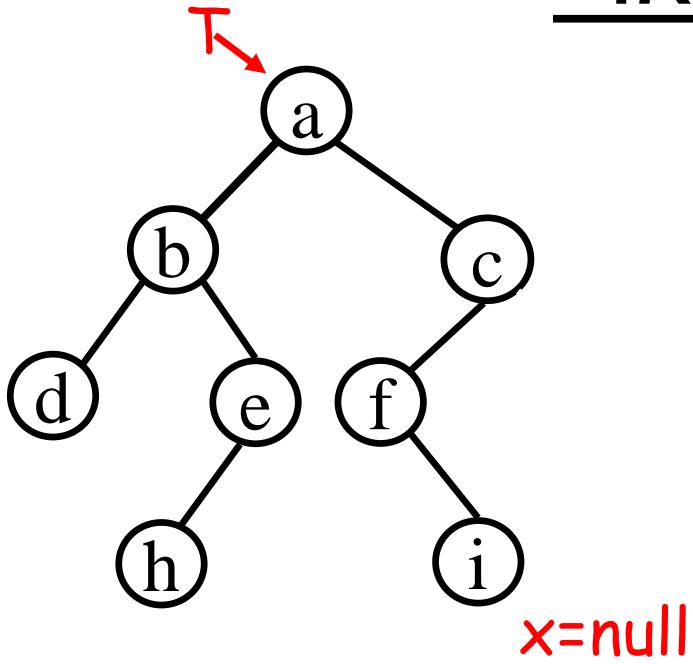
סrox את תת העץ השמאלי

**בקר בשורש**

סrox את תת העץ הימני

הדפסת כל מפתחות העץ בסדר inorder

# סרייה עץ בינארי



d b h e a f i

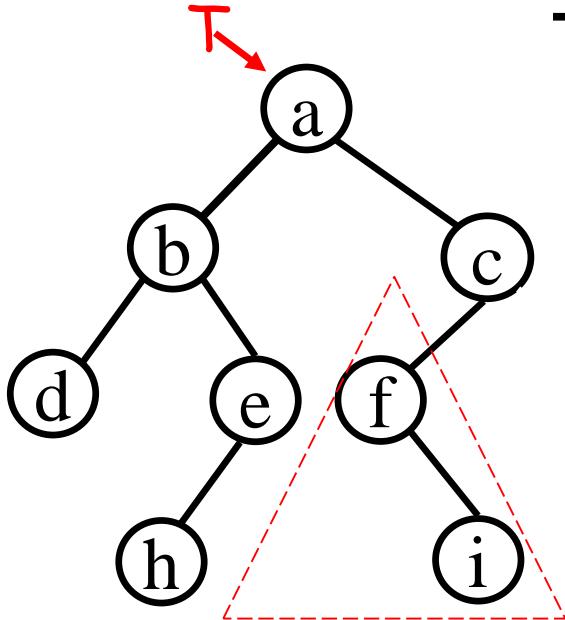
**סרייה תואנית inorder**

סrox את תת העץ השמאלי  
בקר בשורש

סrox את תת העץ הימני

הדפסת כל מפתחות העץ בסדר inorder

# סרייה עץ בינארי



d b h e a f i

## סרייה תומכית **inorder**

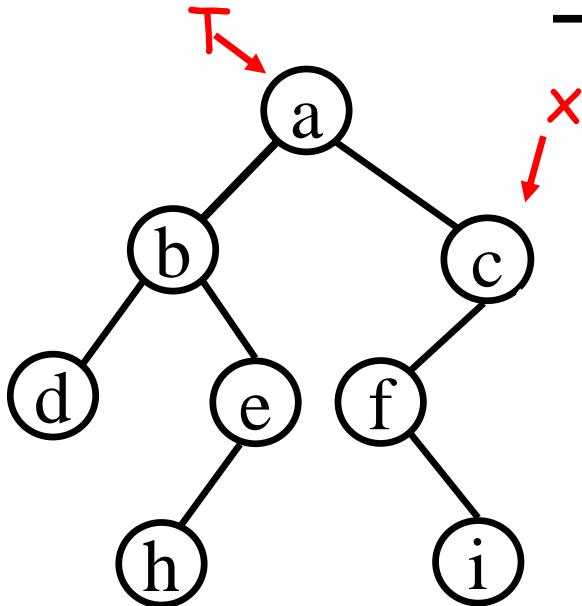
סrox את תת העץ השמאלי

בקיר בשורש

סrox את תת העץ הימני

הדפסת כל מפתחות העץ בסדר **inorder**

# סרייה עץ בינארי



d b h e a f i c

**סרייה תומכת inorder**

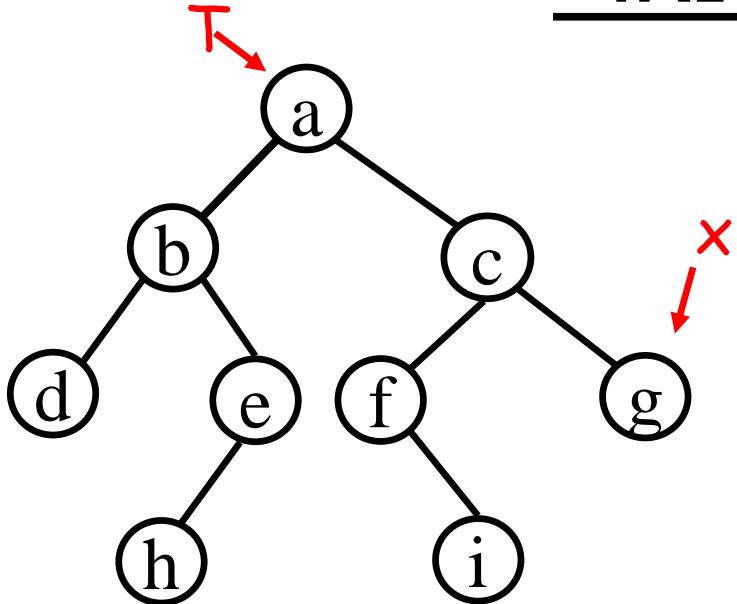
סrox את תת העץ השמאלי

**בקר בשורש**

סrox את תת העץ הימני

הדפסת כל מפתחות העץ בסדר inorder

# סרייה עץ בינארי



d b h e a f i c

**סרייה תואנית inorder**

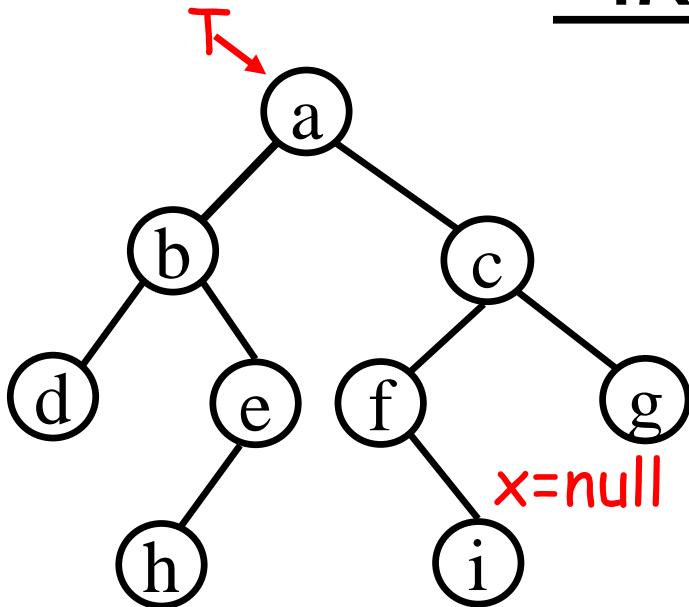
סrox את תת העץ השמאלי

בקר בשורש

**סrox את תת העץ ימני**

הדפסת כל מפתחות העץ בסדר inorder

# סרייה עץ בינארי



d b h e a f i c

**סרייה תואנית inorder**

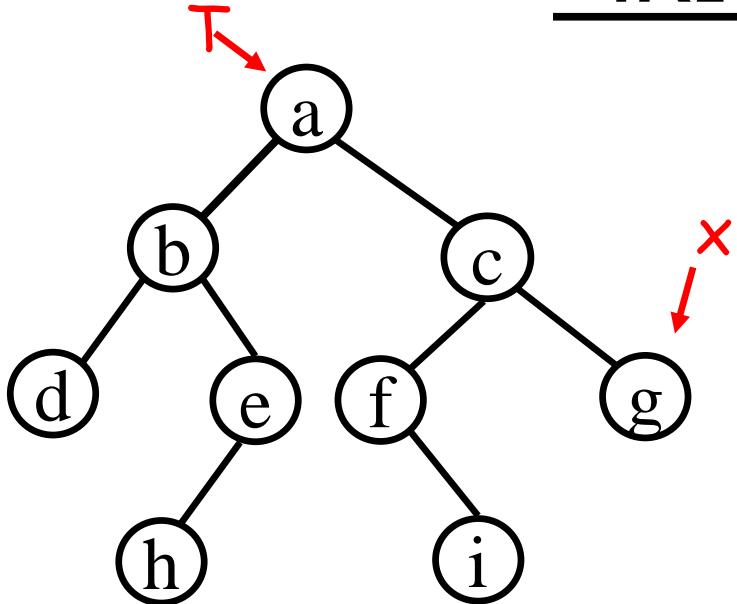
סrox את תת העץ השמאלי

בקר בשורש

סrox את תת העץ הימני

הדפסת כל מפתחות העץ בסדר inorder

# סרייה עץ בינארי



d b h e a f i c g

**סרייה תומכית inorder**

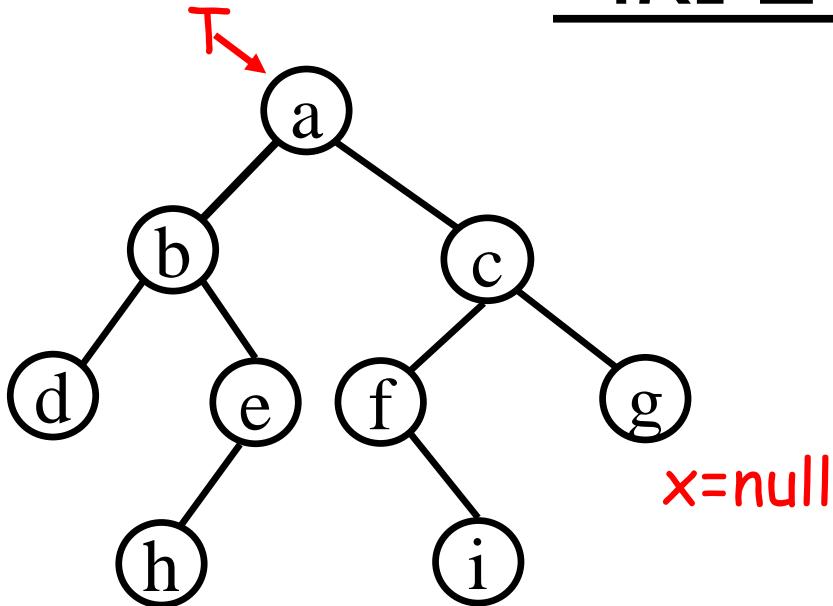
סrox את תת העץ השמאלי

**בקר בשורש**

סrox את תת העץ הימני

הדפסת כל מפתחות העץ בסדר inorder

# סרייה עץ בינארי



d b h e a f i c g

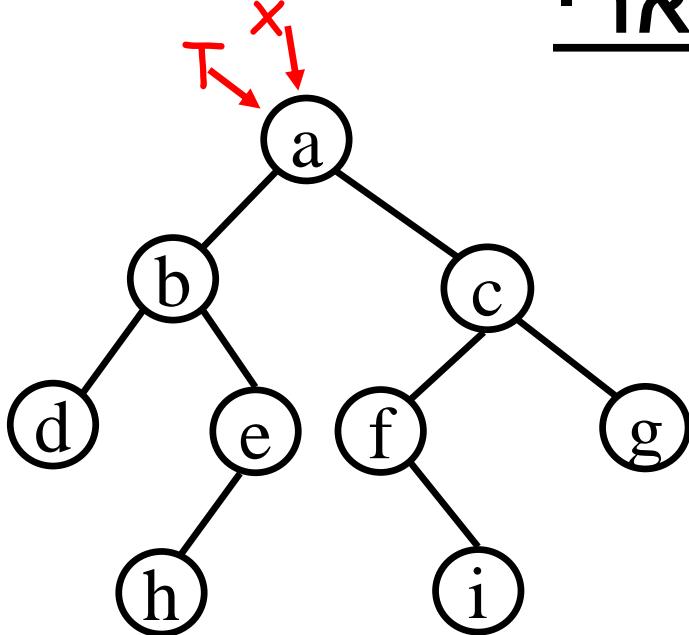
**סרייה תואנית inorder**

סrox את תת העץ השמאלי  
בקר בשורש

סrox את תת העץ הימני

הדפסת כל מפתחות העץ בסדר inorder

# סרייה עץ בינארי



d b h e a f i c g

## סרייה תואנית **inorder**

סrox את תת העץ השמאלי

בקר בשורש

סrox את תת העץ הימני

הדפסת כל מפתחות העץ בסדר **inorder**

**inorder-print(x)**

**if**  $x \neq \text{null}$  **then**

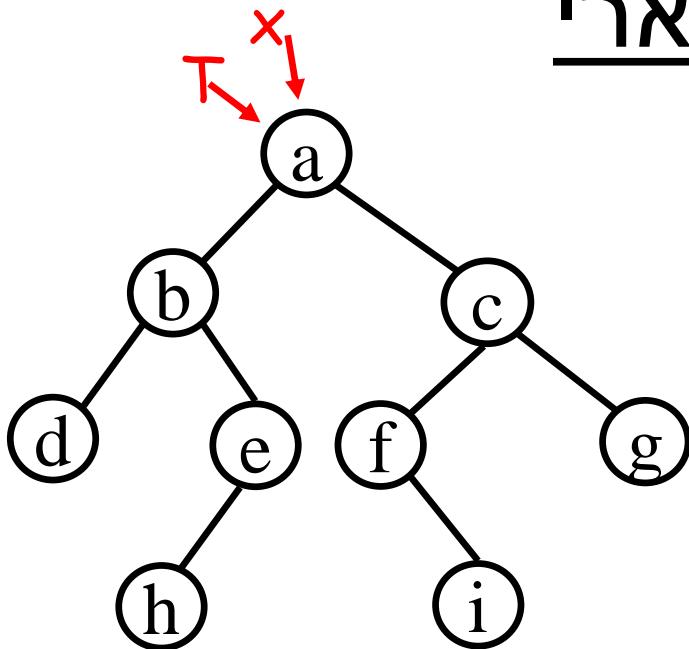
**inorder-print(left(x))**

**print key(x)**

**inorder-print(right(x))**

קריאה: **inorder-print(root(T))**  
סיבוכיות זמן:  $(n) \Theta$  כאשר  $n$  הוא מספר הצמתים בעץ

# סרייה עץ בינארי



**סרייה תחילה בker בשורש**

סרווק את תת העץ השמאלי

סרווק את תת העץ הימני

הדפסת כל מפתחות העץ בסדר inorder

**preorder-print(x)**

**if**  $x \neq \text{null}$  **then**

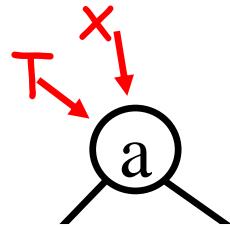
print key( $x$ )

preorder-print(left( $x$ ))

preorder-print(right( $x$ ))

קריאה:  $\text{preorder-print}(\text{root}(T))$   
סיבוכיות זמן:  $(n) \Theta$  כאשר  $n$  הוא מספר הצמתים בעץ

# סרייה עץ בינארי



**סרייה תחילה**  
**בקר בשורש**

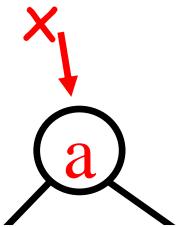
סrox את תת העץ השמאלי  
סrox את תת העץ הימני

הדפסת כל מפתחות העץ בסדר **inorder**

```
preorder-print(x)
if x ≠ null then
    print key(x)
    preorder-print(left(x))
    preorder-print(right(x))
```

הקריאה:  $((T))$  **preorder-print(root(T))**  
סיבוכיות זמן:  $(n)$   $\Theta$  כאשר  $n$  הוא מספר הצמתים בעץ

# סרייה עץ בינארי



**סרייה תחילה** **preorder**  
**בקר בשורש**

סrox את תת העץ השמאלי

סrox את תת העץ הימני

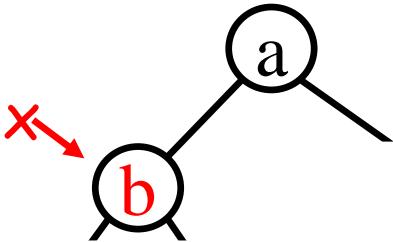
הדפסת כל מפתחות העץ בסדר **inorder**

**preorder-print(x)**  
if  $x \neq \text{null}$  then  
    print key( $x$ )  
    preorder-print(left( $x$ ))  
    preorder-print(right( $x$ ))

a

הקריאה:  $\text{preorder-print}(\text{root}(T))$   
סיבוכיות זמן:  $(n) \Theta$  כאשר  $n$  הוא מספר הצמתים בעץ

# סרייה עץ בינארי



**סרייה תחילה preorder**

בקר בשורש

סrox את תת העץ השמאלי

סrox את תת העץ הימני

הדפסת כל מפתחות העץ בסדר inorder

preorder-print( $x$ )

if  $x \neq \text{null}$  then

print key( $x$ )

preorder-print(left( $x$ ))

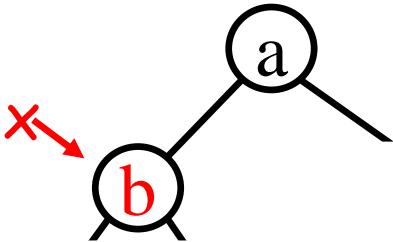
preorder-print(right( $x$ ))

a

הקריאה:  $((T)) \text{root} \text{preorder-print}$

סיבוכיות זמן:  $(n) \Theta$  כאשר  $\Theta$  הוא מספר הצמתים בעץ

# סרייה עץ בינארי



**סרייה תחילה preorder**

**ברק בשורש**

סrox את תת העץ השמאלי

סrox את תת העץ הימני

הדפסת כל מפתחות העץ בסדר inorder

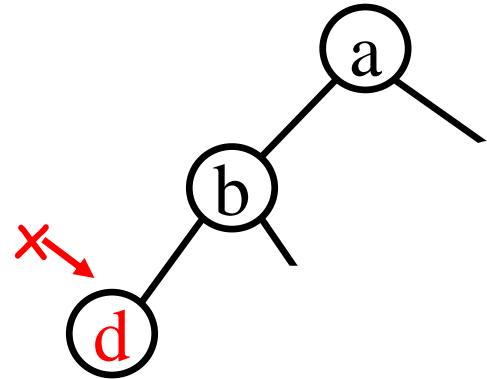
a b

```

preorder-print(x)
if x ≠ null then
    print key(x)
    preorder-print(left(x))
    preorder-print(right(x))
  
```

הקריאה:  $\text{preorder-print}(\text{root}(T))$   
סיבוכיות זמן:  $O(n)$  כאשר  $n$  הוא מספר הצמתים בעץ

# סרייה עץ בינארי



**סדרה תחילה preorder**

בקר בשורש

סrox את תת העץ השמאלי

סrox את תת העץ הימני

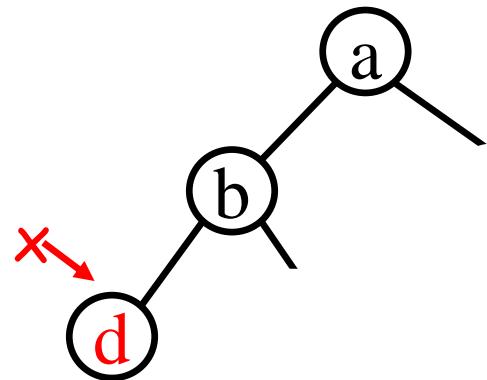
הdfs כל מפתחות העץ בסדר inorder

a b

```
preorder-print(x)
if x ≠ null then
    print key(x)
    preorder-print(left(x))
    preorder-print(right(x))
```

הקריאה: ((T))root().preorder-print(  
סיבוכיות זמן:  $\Theta(n)$  כאשר  $n$  הוא מספר הצמתים בעץ

# סרייה עץ בינארי



**סרייה תחילה**

**בקר בשורש**

סrox את תת העץ השמאלי

סrox את תת העץ הימני

הדפסת כל מפתחות העץ בסדר **inorder**

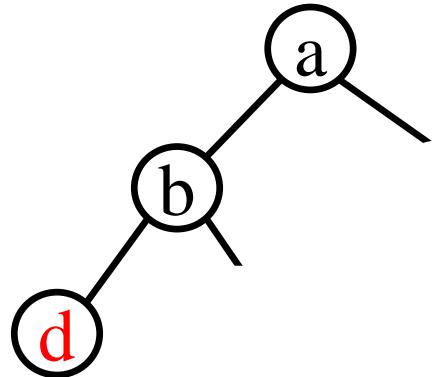
a b d

```

preorder-print(x)
if x ≠ null then
    print key(x)
    preorder-print(left(x))
    preorder-print(right(x))
  
```

קריאה: `((T))root().preorder-print()`  
סיבוכיות זמן:  $O(n)$  כאשר  $n$  הוא מספר הצמתים בעץ

# סרייה עץ בינארי



**סרייה תחילה preorder**

בקר בשורש

סrox את תת העץ השמאלי

סrox את תת העץ הימני

הדפסת כל מפתחות העץ בסדר inorder

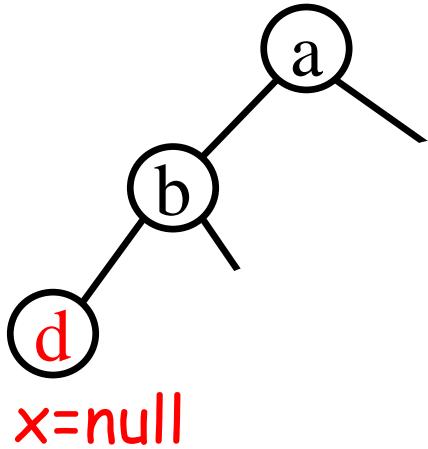
a b d

```

preorder-print(x)
if x ≠ null then
    print key(x)
    preorder-print(left(x))
    preorder-print(right(x))
  
```

קריאה:  $\text{preorder-print}(\text{root}(T))$   
סיבוכיות זמן:  $O(n)$  כאשר  $n$  הוא מספר הצמתים בעץ

# סרייה עץ בינארי



**סדרה תחילה** **preorder**

בקר בשורש

סrox את תת העץ השמאלי

סrox את תת העץ הימני

הdfsת כל מפתחות העץ בסדר **inorder**

a b d

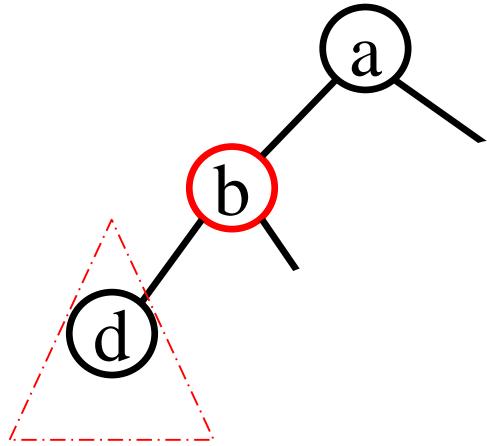
```

preorder-print(x)
if x ≠ null then
    print key(x)
    preorder-print(left(x))
    preorder-print(right(x))

```

הקריאה: ((T))root().preorder-print()  
סיבוכיות זמן:  $O(n)$  כאשר  $n$  הוא מספר הצמתים בעץ

# סרייה עץ בינארי



## סרייה תחילה preorder

בקר בשורש

סrox את תת העץ השמאלי

סrox את תת העץ הימני

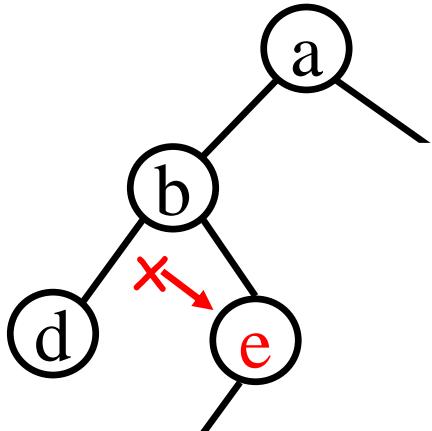
הדפסת כל מפתחות העץ בסדר inorder

a b d

```
preorder-print(x)
if x ≠ null then
    print key(x)
    preorder-print(left(x))
    preorder-print(right(x))
```

קריאה:  $((T)) \text{root} \text{preorder-print}$   
סיבוכיות זמן:  $(n) \Theta$  כאשר  $n$  הוא מספר הצמתים בעץ

# סרייה עץ בינארי



**סרייה תחילה preorder**

בקר בשורש

סrox את תת העץ השמאלי

סrox את תת העץ הימני

הדפסת כל מפתחות העץ בסדר inorder

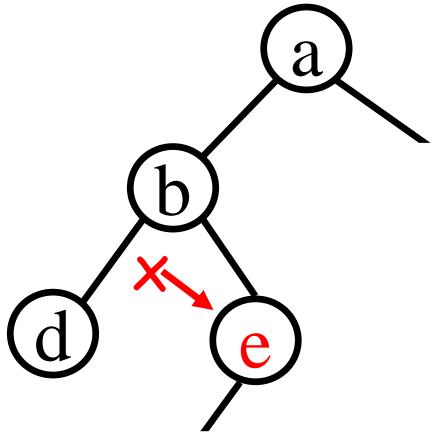
a b d

```

preorder-print(x)
if x ≠ null then
    print key(x)
    preorder-print(left(x))
    preorder-print(right(x))
  
```

הקריאה:  $\text{preorder-print}(\text{root}(T))$   
סיבוכיות זמן:  $O(n)$  כאשר  $n$  הוא מספר הצמתים בעץ

# סרייה עץ בינארי



**סרייה תחילה preorder**

**ברק בשורש**

סrox את תת העץ השמאלי

סrox את תת העץ הימני

הדפסת כל מפתחות העץ בסדר inorder

a b d e

**preorder-print(x)**

**if**  $x \neq \text{null}$  **then**

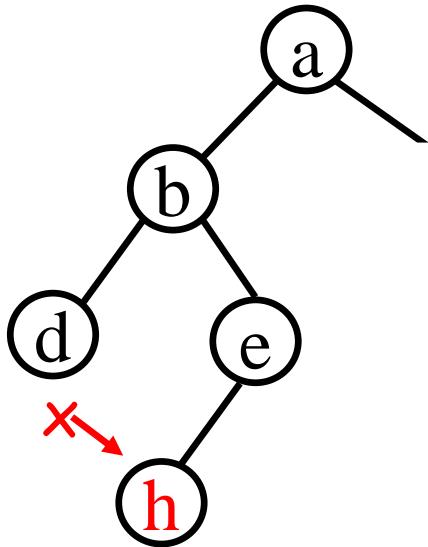
print key( $x$ )

preorder-print(left( $x$ ))

preorder-print(right( $x$ ))

קריאה:  $\text{preorder-print}(\text{root}(T))$   
סיבוכיות זמן:  $(n) \Theta$  כאשר  $n$  הוא מספר הצמתים בעץ

# סרייה עץ בינארי



a b d e

**סדרה תחילה**

בקר בשורש

סrox את תת העץ השמאלי

סrox את תת העץ הימני

הdfs כל מפתחות העץ בסדר inorder

**preorder-print(x)**

**if**  $x \neq \text{null}$  **then**

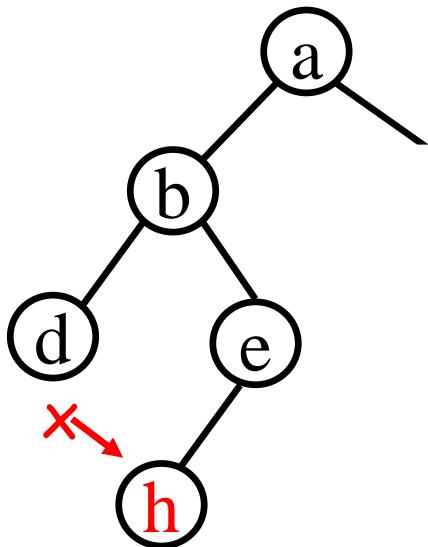
print key( $x$ )

preorder-print(left( $x$ ))

preorder-print(right( $x$ ))

קריאה:  $\text{preorder-print}(\text{root}(T))$   
סיבוכיות זמן:  $(n) \Theta$  כאשר  $n$  הוא מספר הצמתים בעץ

# סרייה עץ בינארי



a b d e h

**סרייה תחילה בroot**

סrox את תת העץ השמאלי

סrox את תת העץ הימני

הדפסת כל מפתחות העץ בסדר inorder

**preorder-print(x)**

**if**  $x \neq \text{null}$  **then**

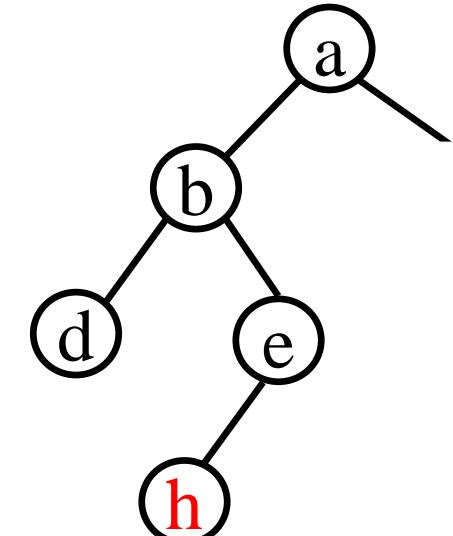
print key( $x$ )

preorder-print(left( $x$ ))

preorder-print(right( $x$ ))

קריאה:  $\text{preorder-print}(\text{root}(T))$   
סיבוכיות זמן:  $(n) \Theta$  כאשר  $n$  הוא מספר הצמתים בעץ

# סרייה עץ בינארי



$x=null$   
a b d e h

**סדרה תחילה preorder**

בקר בשורש

סrox את תת העץ השמאלי

סrox את תת העץ הימני

הדפסת כל מפתחות העץ בסדר inorder

preorder-print( $x$ )

if  $x \neq null$  then

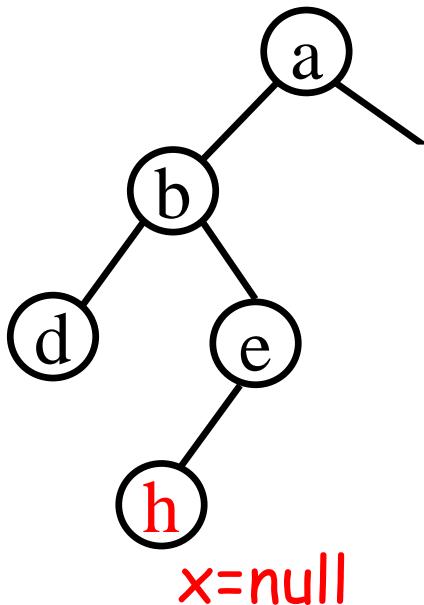
print key( $x$ )

preorder-print(left( $x$ ))

preorder-print(right( $x$ ))

קריאה:  $((T))root$  סיבוכיות זמן:  $(n) \Theta$  כאשר  $n$  הוא מספר הצמתים בעץ

סרייקת עץ בינהARI

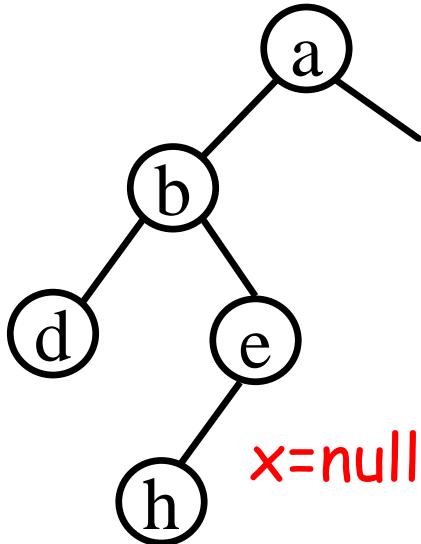


a b d e h

```
preorder-print(x)
if x ≠ null then
    print key(x)
    preorder-print(left(x))
    preorder-print(right(x))
```

הקריאה: `preorder-print(root(T))`  
 סיבוכיות זמן:  $O(n)$  כאשר  $n$  הוא מספר הצלמתים בעץ

# סרייה עץ בינארי



a b d e h

**סדרה תחילה**

בקר בשורש

סrox את תת העץ השמאלי

סrox את תת העץ הימני

הדפסת כל מפתחות העץ בסדר inorder

preorder-print( $x$ )

if  $x \neq \text{null}$  then

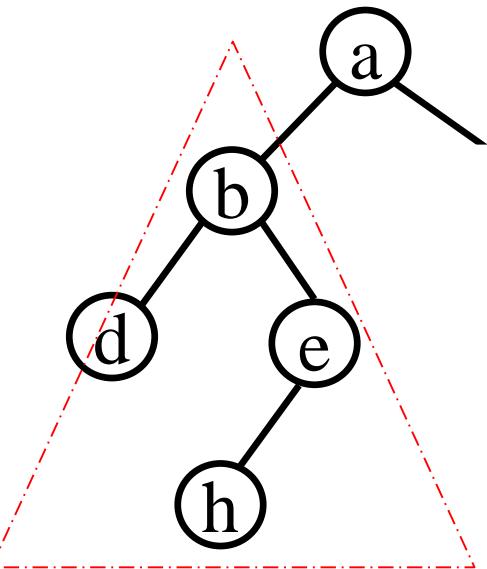
print key( $x$ )

preorder-print(left( $x$ ))

preorder-print(right( $x$ ))

קריאה:  $\text{preorder-print}(\text{root}(T))$   
סיבוכיות זמן:  $(n) \Theta$  כאשר  $n$  הוא מספר הצמתים בעץ

# סרייה עץ בינארי



a b d e h

## סרייה תחילה preorder

בקר בשורש

סrox את תת העץ השמאלי

סrox את תת העץ הימני

הדפסת כל מפתחות העץ בסדר inorder

preorder-print( $x$ )

if  $x \neq \text{null}$  then

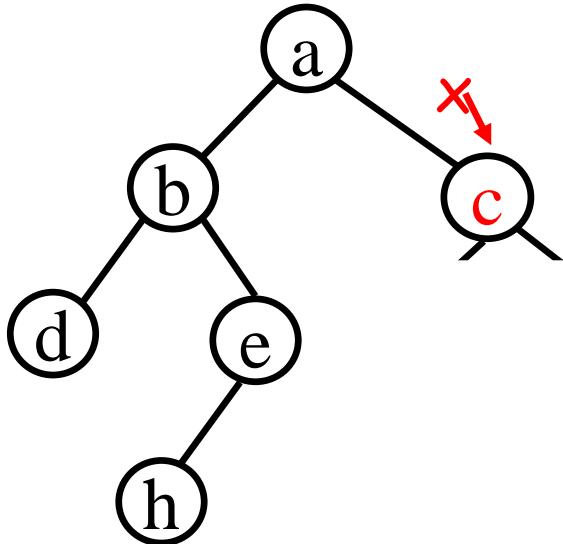
print key( $x$ )

preorder-print(left( $x$ ))

preorder-print(right( $x$ ))

קריאה:  $\text{preorder-print}(\text{root}(T))$   
סיבוכיות זמן:  $(n) \Theta$  כאשר  $n$  הוא מספר הצמתים בעץ

# סרייה עץ בינארי



a b d e h

## סרייה תחילה preorder

בקר בשורש

סrox את תת העץ השמאלי

סrox את תת העץ הימני

הדפסת כל מפתחות העץ בסדר inorder

preorder-print( $x$ )

if  $x \neq \text{null}$  then

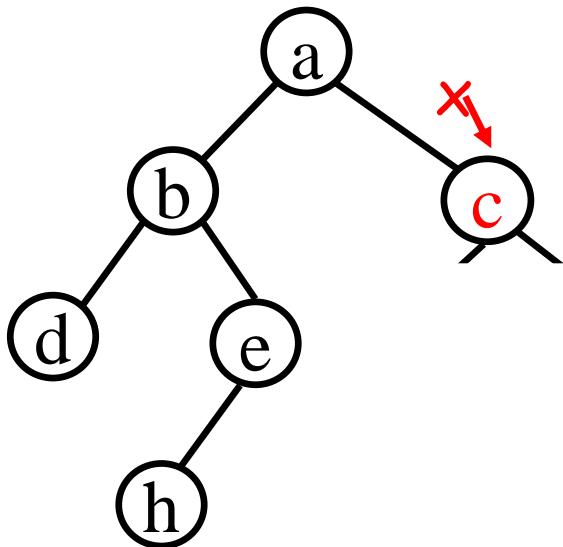
print key( $x$ )

preorder-print(left( $x$ ))

preorder-print(right( $x$ ))

קריאה:  $\text{preorder-print}(\text{root}(T))$   
סיבוכיות זמן:  $(n) \Theta$  כאשר  $n$  הוא מספר הצמתים בעץ

## סרייקת עץ בינהARI



a b d e h c

# סְרִיקָה תְּחִילִית pre-order בָּקָר בְּשׂוֹרֶשֶׁ

## סורך את תת העז השמאלי

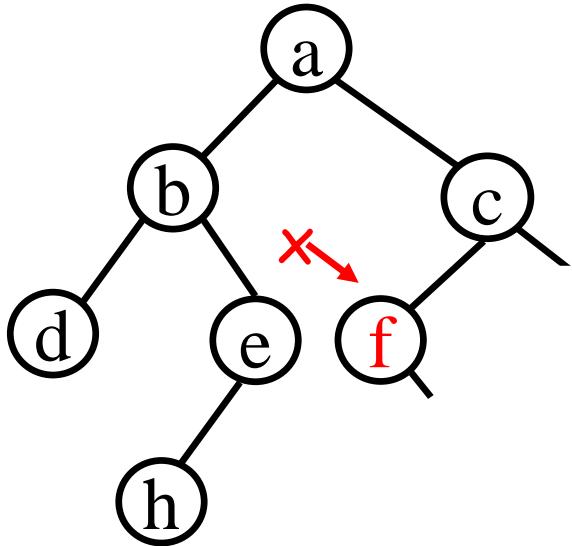
## סrox את Tat ha'ez ha'imani

הדף כל מפתחות העץ בסדר inorder

```
preorder-print(x)
if x ≠ null then
    print key(x)
    preorder-print(left(x))
    preorder-print(right(x))
```

הקריאה: `preorder-print(root(T))`  
 סיבוכיות זמן:  $O(n)$  כאשר  $n$  הוא מספר הצלמתים בעץ

# סרייה עץ בינארי



a b d e h c

**סדרה תחילה preorder**

בקר בשורש

סrox את תת העץ השמאלי

סrox את תת העץ הימני

הדפסת כל מפתחות העץ בסדר inorder

preorder-print( $x$ )

if  $x \neq \text{null}$  then

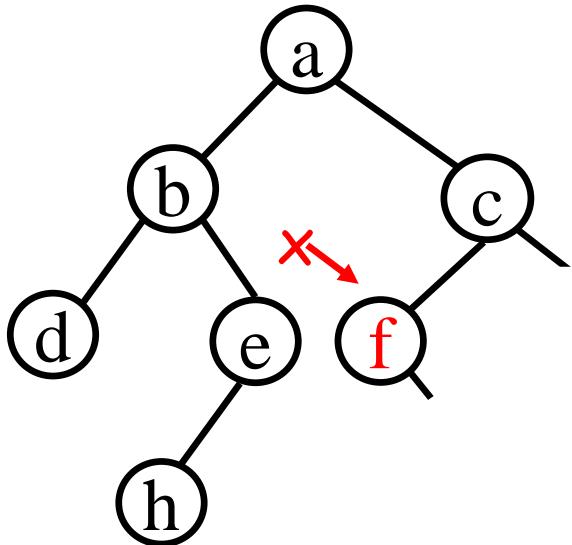
print key( $x$ )

preorder-print(left( $x$ ))

preorder-print(right( $x$ ))

קריאה:  $\text{preorder-print}(\text{root}(T))$   
סיבוכיות זמן:  $(n) \Theta$  כאשר  $n$  הוא מספר הצמתים בעץ

# סרייה עץ בינארי



a b d e h c f

**סרייה תחילה בroot**

סrox את תת העץ השמאלי

סrox את תת העץ הימני

הדפסת כל מפתחות העץ בסדר inorder

**preorder-print(x)**

**if**  $x \neq \text{null}$  **then**

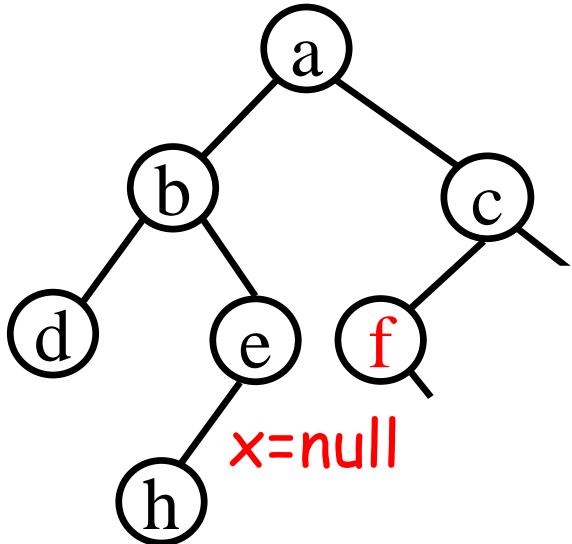
print key( $x$ )

preorder-print(left( $x$ ))

preorder-print(right( $x$ ))

קריאה:  $\text{preorder-print}(\text{root}(T))$   
סיבוכיות זמן:  $(n) \Theta$  כאשר  $n$  הוא מספר הצמתים בעץ

# סרייה עץ בינארי



a b d e h c f

**סדרה תחילה preorder**

בקר בשורש

סrox את תת העץ השמאלי

סrox את תת העץ הימני

הדפסת כל מפתחות העץ בסדר inorder

preorder-print( $x$ )

if  $x \neq \text{null}$  then

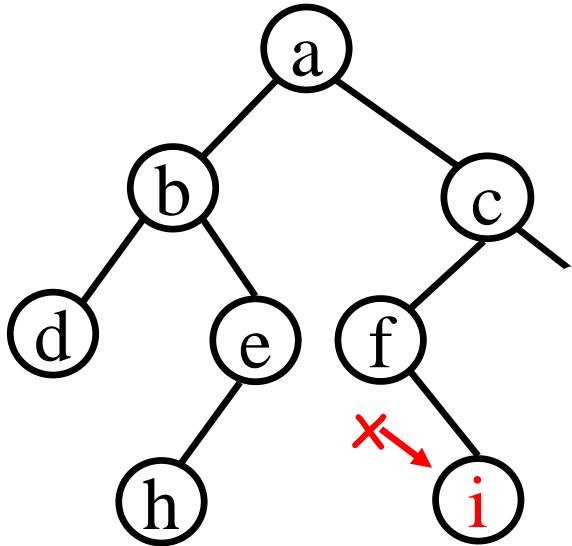
print key( $x$ )

preorder-print(left( $x$ ))

preorder-print(right( $x$ ))

קריאה:  $\text{preorder-print}(\text{root}(T))$   
סיבוכיות זמן:  $(n) \Theta$  כאשר  $n$  הוא מספר הצמתים בעץ

# סרייה עץ בינארי



a b d e h c f

## סדרה תחילה preorder

בקר בשורש  
סrox את תת העץ השמאלי  
**סrox את תת העץ הימני**

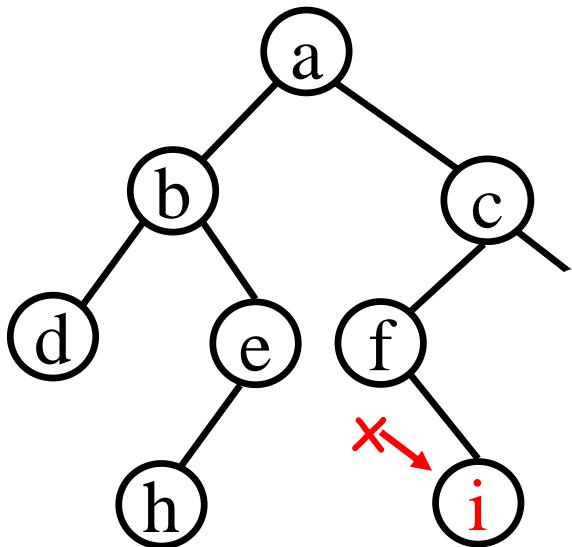
הדפסת כל מפתחות העץ בסדר inorder

```

preorder-print(x)
if x ≠ null then
    print key(x)
    preorder-print(left(x))
    preorder-print(right(x))
  
```

קריאה:  $\text{preorder-print}(\text{root}(T))$   
סיבוכיות זמן:  $O(n)$  כאשר  $n$  הוא מספר הצמתים בעץ

# סרייה עץ בינארי



a b d e h c f i

**סרייה תחילה בroot**

סrox את תת העץ השמאלי

סrox את תת העץ הימני

הדפסת כל מפתחות העץ בסדר inorder

preorder-print( $x$ )

if  $x \neq \text{null}$  then

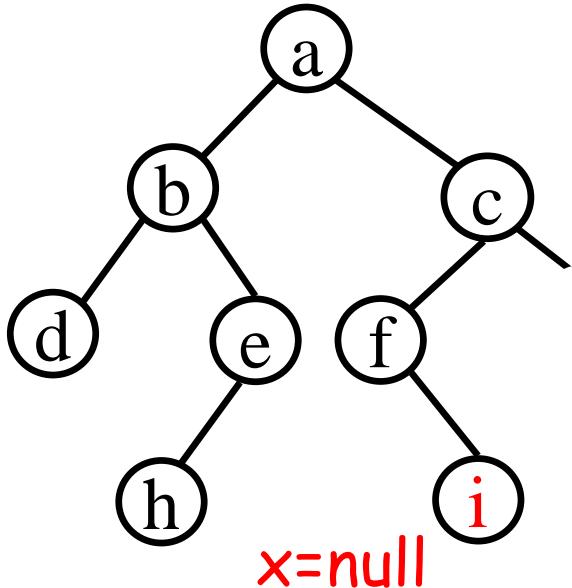
print key( $x$ )

preorder-print(left( $x$ ))

preorder-print(right( $x$ ))

קריאה:  $\text{preorder-print}(\text{root}(T))$   
סיבוכיות זמן:  $(n) \Theta$  כאשר  $n$  הוא מספר הצמתים בעץ

# סרייה עץ בינארי



a b d e h c f i

**סדרה תחילה preorder**

בקר בשורש

סrox את תת העץ השמאלי

סrox את תת העץ הימני

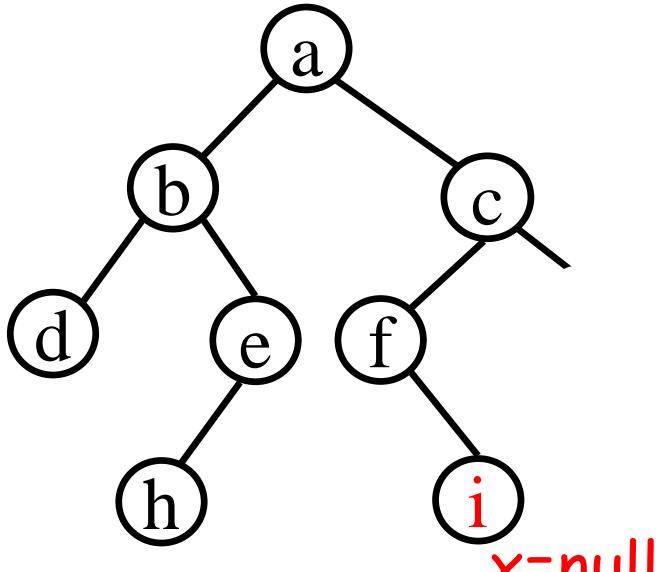
הדפסת כל מפתחות העץ בסדר inorder

```

preorder-print(x)
if x ≠ null then
    print key(x)
    preorder-print(left(x))
    preorder-print(right(x))
  
```

הקריאה: `((T)root) preorder-print`  
סיבוכיות זמן:  $(n)\Theta$  כאשר  $\Theta$  הוא מספר הצמתים בעץ

# סרייה עץ בינארי



a b d e h c f i

## סדרה תחילה preorder

בקר בשורש

סrox את תת העץ השמאלי

סrox את תת העץ הימני

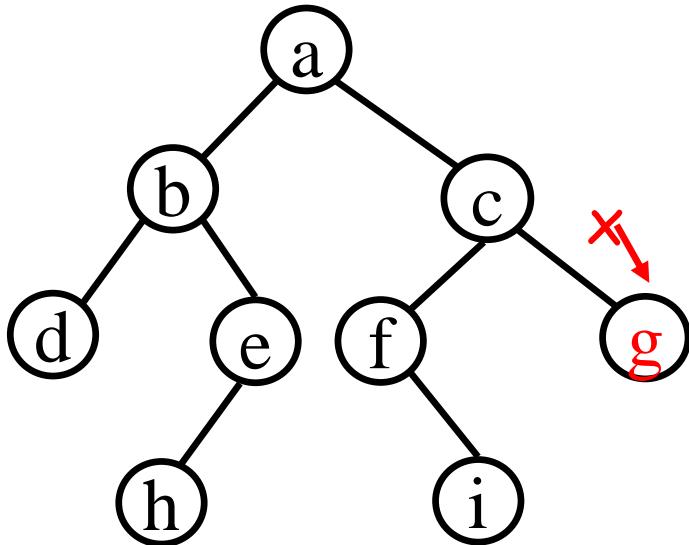
הדפסת כל מפתחות העץ בסדר inorder

```

preorder-print(x)
if x ≠ null then
    print key(x)
    preorder-print(left(x))
    preorder-print(right(x))
  
```

קריאה:  $\text{preorder-print}(\text{root}(T))$   
סיבוכיות זמן:  $O(n)$  כאשר  $n$  הוא מספר הצמתים בעץ

# סרייה עץ בינארי



a b d e h c f i

## סדרה תחילה preorder

בקר בשורש

סrox את תת העץ השמאלי

סrox את תת העץ הימני

הדפסת כל מפתחות העץ בסדר inorder

preorder-print( $x$ )

if  $x \neq \text{null}$  then

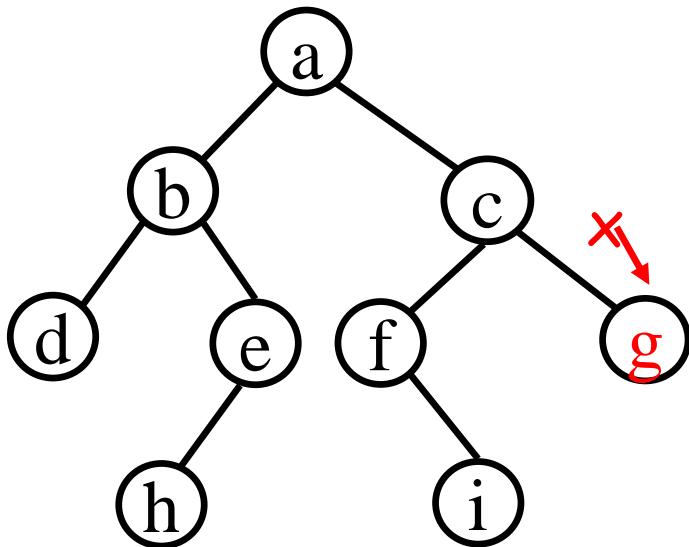
print key( $x$ )

preorder-print(left( $x$ ))

preorder-print(right( $x$ ))

קריאה:  $\text{preorder-print}(\text{root}(T))$   
סיבוכיות זמן:  $(n) \Theta$  כאשר  $n$  הוא מספר הצמתים בעץ

# סרייה עץ בינארי



a b d e h c f i g

**סרייה תחילה בroot**

סrox את תת העץ השמאלי

סrox את תת העץ הימני

הדפסת כל מפתחות העץ בסדר inorder

preorder-print( $x$ )

if  $x \neq \text{null}$  then

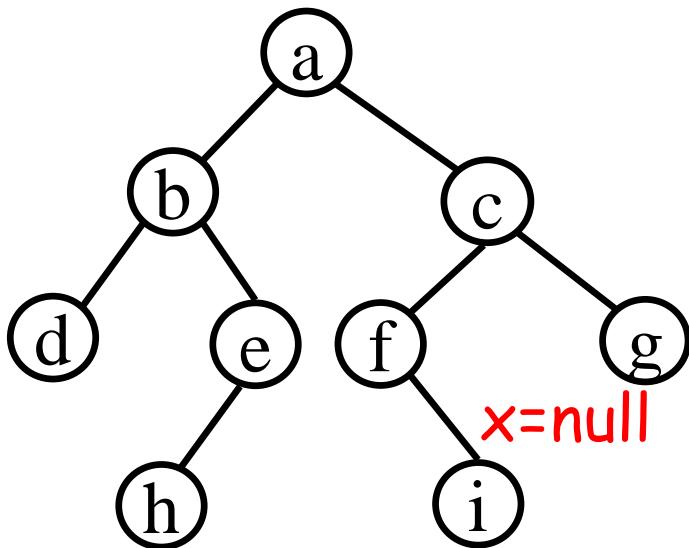
print key( $x$ )

preorder-print(left( $x$ ))

preorder-print(right( $x$ ))

קריאה:  $\text{preorder-print}(\text{root}(T))$   
סיבוכיות זמן:  $(n) \Theta$  כאשר  $n$  הוא מספר הצמתים בעץ

## סיקת עץ בינהי

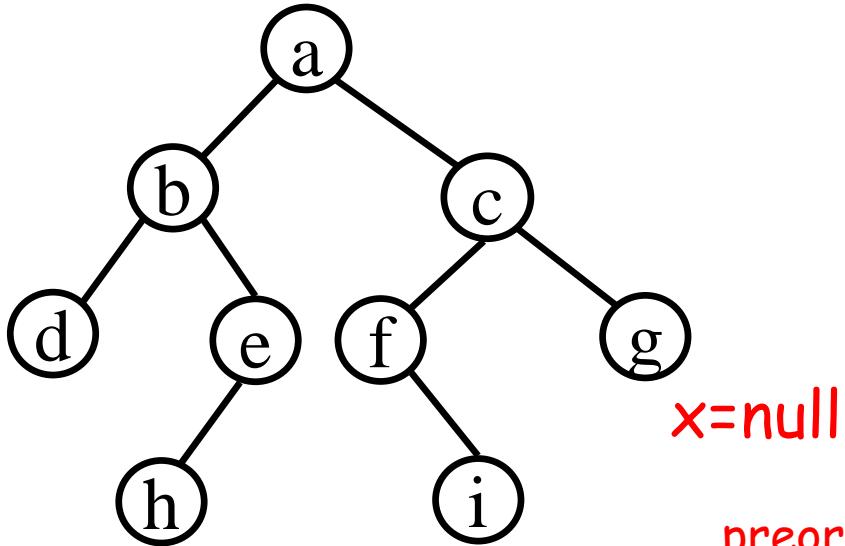


a b d e h c f i g

```
preorder-print(x)
if x ≠ null then
    print key(x)
    preorder-print(left(x))
    preorder-print(right(x))
```

הקריאה: `preorder-print(root(T))`  
 סיבוכיות זמן:  $(n) \Theta$  כאשר  $\Theta$  הוא מספר הצלמתים בעץ

# סרייה עץ בינארי



a b d e h c f i g

**סדרה תחילה**

ברק בשורש

סרוק את תת העץ השמאלי

סרוק את תת העץ הימני

הדפסת כל מפתחות העץ בסדר inorder

preorder-print( $x$ )

if  $x \neq \text{null}$  then

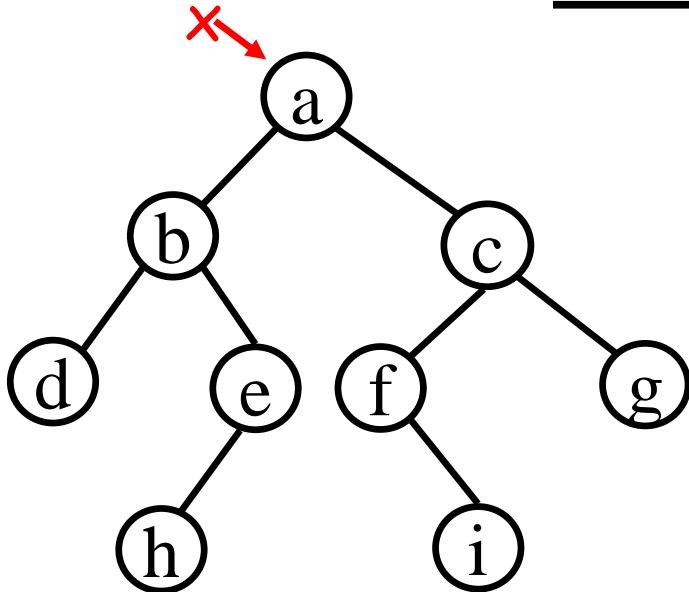
print key( $x$ )

preorder-print(left( $x$ ))

preorder-print(right( $x$ ))

קריאה:  $\text{preorder-print}(\text{root}(T))$   
סיבוכיות זמן:  $(n) \Theta$  כאשר  $n$  הוא מספר הצמתים בעץ

# סרייה עץ בינארי



## סדרה סופית postorder

סורך את תת העץ השמאלי

סורך את תת העץ הימני

בקר בשורש

הדפסת כל מפתחות העץ בסדר inorder

inorder-print( $x$ )

if  $x \neq \text{null}$  then

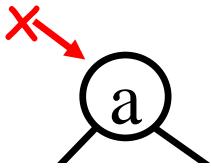
postorder-print(left( $x$ ))

postorder-print(right( $x$ ))

print key( $x$ )

קריאה: postorder-print(root( $T$ ))  
סיבוכיות זמן:  $\Theta(n)$  כאשר  $n$  הוא מספר הצמתים בעץ

# סרייה עץ בינארי



## סרייה סופית postorder

סורך את תת העץ השמאלי

סורך את תת העץ הימני

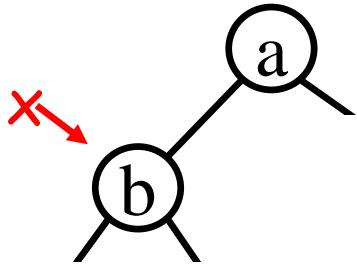
בקר בשורש

הדפסת כל מפתחות העץ בסדר inorder

```
inorder-print(x)
if x ≠ null then
    postorder-print(left(x))
    postorder-print(right(x))
    print key(x)
```

קריאה: `postorder-print(root(T))`  
סיבוכיות זמן:  $\Theta(n)$  כאשר  $n$  הוא מספר הצמתים בעץ

# סרייה עץ בינארי



**סרייה סופית העץ**

**postorder**  
סrox את תת העץ השמאלי

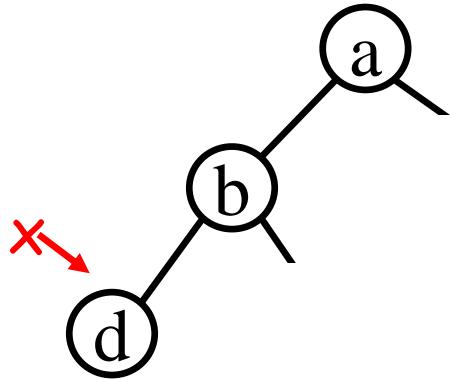
סrox את תת העץ הימני  
בקר בשורש

הדפסת כל מפתחות העץ בסדר **inorder**

```
inorder-print(x)
if x ≠ null then
    postorder-print(left(x))
    postorder-print(right(x))
    print key(x)
```

קריאה: **postorder-print(root(T))**  
סיבוכיות זמן:  $(n)\Theta$  כאשר  $\Theta$  הוא מספר הצמתים בעץ

# סרייה עץ בינארי



**סרייה סופית postorder**

סורך את תת העץ השמאלי

סורך את תת העץ הימני

בקר בשורש

הדפסת כל מפתחות העץ בסדר inorder

inorder-print( $x$ )

if  $x \neq \text{null}$  then

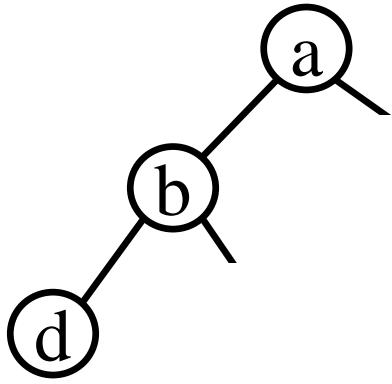
postorder-print(left( $x$ ))

postorder-print(right( $x$ ))

print key( $x$ )

קריאה:  $\text{postorder-print}(\text{root}(T))$   
סיבוכיות זמן:  $(n)\Theta$  כאשר  $\Theta$  הוא מספר הצמתים בעץ

# סרייה עץ בינארי



x=null

## סרייה סופית postorder

סורך את תת העץ השמאלי

סורך את תת העץ הימני

בקר בשורש

הדפסת כל מפתחות העץ בסדר inorder

inorder-print(x)

if  $x \neq \text{null}$  then

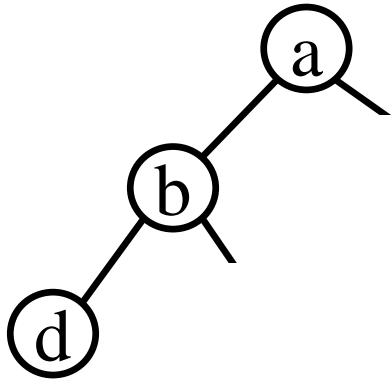
postorder-print(left( $x$ ))

postorder-print(right( $x$ ))

print key( $x$ )

קריאה: `postorder-print(root(T))`  
סיבוכיות זמן:  $(n)\Theta$  כאשר  $\Theta$  הוא מספר הצמתים בעץ

# סרייה עץ בינארי



$x=null$

## סרייה סופית postorder

סrox את תת העץ השמאלי

סrox את תת העץ ימני

בקר בשורש

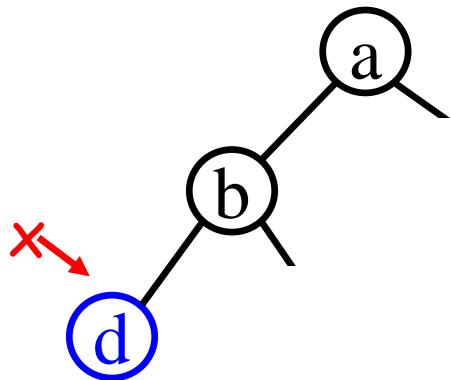
הדפסת כל מפתחות העץ בסדר inorder

```

inorder-print(x)
if x ≠ null then
    postorder-print(left(x))
    postorder-print(right(x))
    print key(x)
  
```

הקריאה:  $\text{postorder-print}(\text{root}(T))$   
סיבוכיות זמן:  $(n)\Theta$  כאשר  $\Theta$  הוא מספר הצמתים בעץ

# סרייה עץ בינארי



## סרייה סופית postorder

סrox את תת העץ השמאלי

סrox את תת העץ הימני

**בקר בשורש**

הדפסת כל מפתחות העץ בסדר inorder

inorder-print( $x$ )

if  $x \neq \text{null}$  then

postorder-print(left( $x$ ))

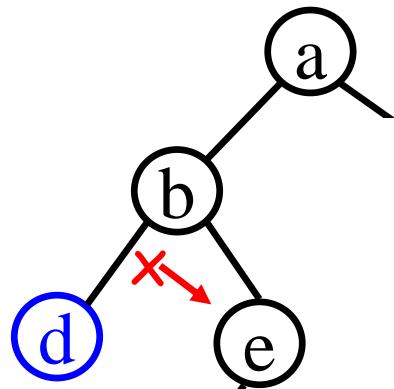
postorder-print(right( $x$ ))

print key( $x$ )

d

קריאה: postorder-print(root( $T$ ))  
סיבוכיות זמן:  $\Theta(n)$  כאשר  $n$  הוא מספר הצמתים בעץ

סרייקת עץ בינהARI



# סְרִיקָה סֹפִית postorder

## סrox את תת העז השמאלי

## סrox את תת העץ הימני

בקר בשורש

הדף כל מפתחות העץ בסדר inorder

**inorder-print(x)**

if  $x \neq \text{null}$  then

postorder-print(left(x))

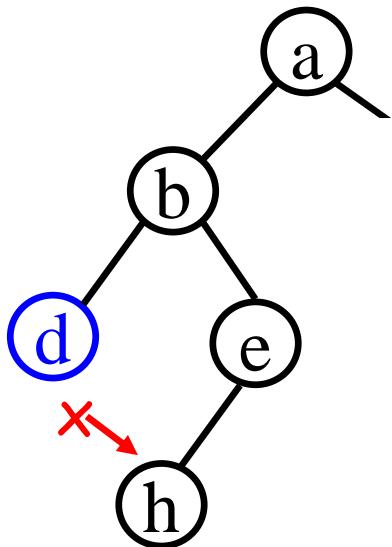
postorder-print(right(x))

print key(x)

d

הקריאה: `postorder-print(root(T))`  
סיבוכיות זמן:  $\Theta(n)$  כאשר  $n$  הוא מספר הצמתים בעץ

# סרייה עץ בינארי



**סרייה סופית postorder**

סורך את תת העץ השמאלי

סורך את תת העץ הימני

בקר בשורש

הדפסת כל מפתחות העץ בסדר inorder

inorder-print( $x$ )

if  $x \neq \text{null}$  then

postorder-print(left( $x$ ))

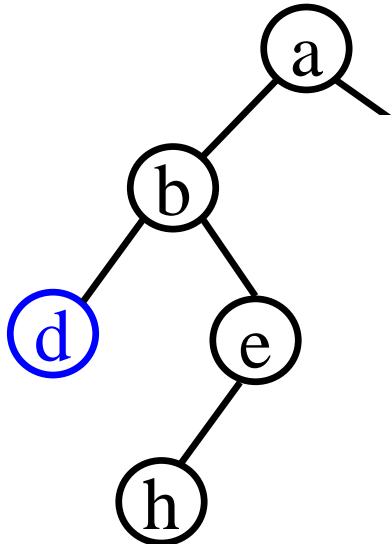
postorder-print(right( $x$ ))

print key( $x$ )

d

קריאה: postorder-print(root( $T$ ))  
סיבוכיות זמן:  $\Theta(n)$  כאשר  $n$  הוא מספר הצמתים בעץ

# סרייה עץ בינארי



x=null

d

## סרייה סופית postorder

סורך את תת העץ השמאלי

סורך את תת העץ הימני

בקר בשורש

הדפסת כל מפתחות העץ בסדר inorder

inorder-print( $x$ )

if  $x \neq \text{null}$  then

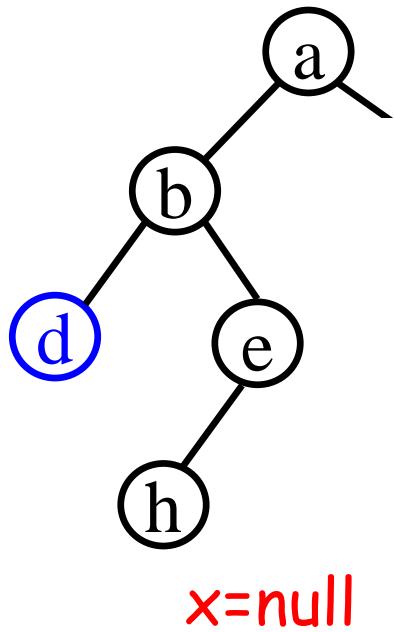
postorder-print(left( $x$ ))

postorder-print(right( $x$ ))

print key( $x$ )

קריאה:  $\text{postorder-print}(\text{root}(T))$   
סיבוכיות זמן:  $(n)\Theta$  כאשר  $n$  הוא מספר הצמתים בעץ

# סרייה עץ בינארי



d

## סרייה סופית postorder

סrox את תת העץ השמאלי

סrox את תת העץ ימני

בקר בשורש

הדפסת כל מפתחות העץ בסדר inorder

inorder-print( $x$ )

if  $x \neq \text{null}$  then

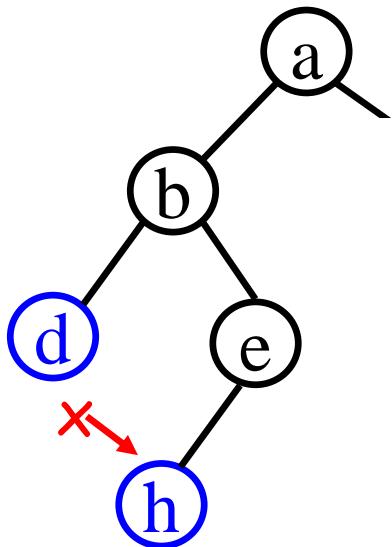
postorder-print(left( $x$ ))

postorder-print(right( $x$ ))

print key( $x$ )

קריאה:  $\text{postorder-print}(\text{root}(T))$   
סיבוכיות זמן:  $(n)\Theta$  כאשר  $n$  הוא מספר הצלמים בעץ

# סרייה עץ בינארי



d h

## סרייה סופית postorder

סrox את תת העץ השמאלי

סrox את תת העץ הימני

**בקר בשורש**

הדפסת כל מפתחות העץ בסדר inorder

inorder-print( $x$ )

if  $x \neq \text{null}$  then

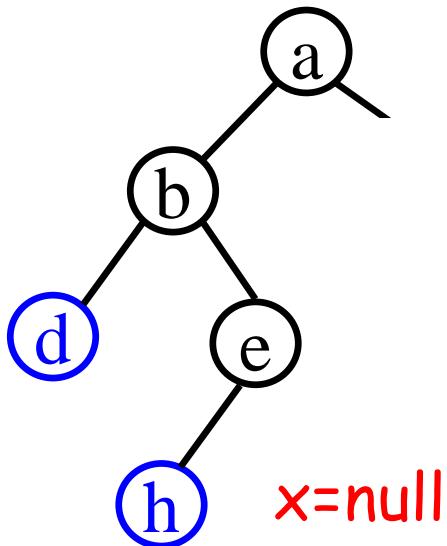
postorder-print(left( $x$ ))

postorder-print(right( $x$ ))

print key( $x$ )

קריאה: `postorder-print(root(T))`  
סיבוכיות זמן:  $\Theta(n)$  כאשר  $n$  הוא מספר הצמתים בעץ

# סרייה עץ בינארי



d h

## סרייה סופית postorder

סrox את תת העץ השמאלי

סrox את תת העץ הימני

בקר בשורש

הדפסת כל מפתחות העץ בסדר inorder

inorder-print( $x$ )

if  $x \neq \text{null}$  then

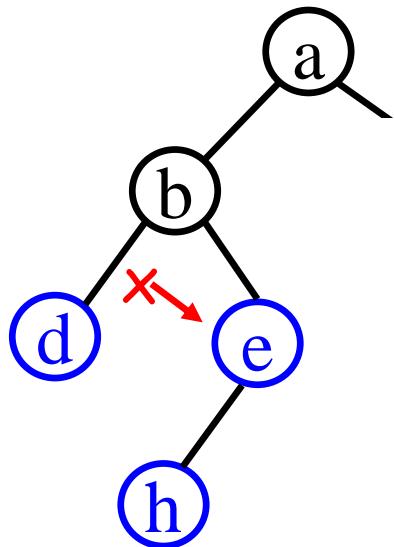
postorder-print(left( $x$ ))

postorder-print(right( $x$ ))

print key( $x$ )

קריאה: `postorder-print(root(T))`  
סיבוכיות זמן:  $\Theta(n)$  כאשר  $n$  הוא מספר הצמתים בעץ

# סרייה עץ בינארי



d h e

## סרייה סופית postorder

סרווק את תת העץ השמאלי

סרווק את תת העץ הימני

**בקר בשורש**

הדפסת כל מפתחות העץ בסדר inorder

inorder-print( $x$ )

if  $x \neq \text{null}$  then

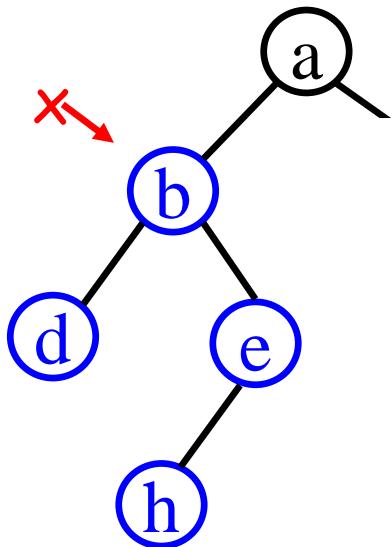
postorder-print(left( $x$ ))

postorder-print(right( $x$ ))

print key( $x$ )

קריאה: `postorder-print(root(T))`  
סיבוכיות זמן:  $\Theta(n)$  כאשר  $n$  הוא מספר הצמתים בעץ

# סרייה עץ בינארי



d h e b

## סדרה סופית postorder

סורך את תת העץ השמאלי

סורך את תת העץ הימני

**בקר בשורש**

הדפסת כל מפתחות העץ בסדר inorder

inorder-print( $x$ )

if  $x \neq \text{null}$  then

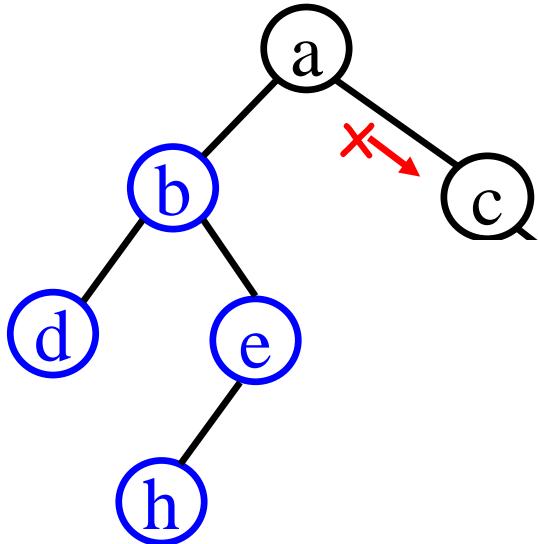
postorder-print(left( $x$ ))

postorder-print(right( $x$ ))

print key( $x$ )

קריאה: `postorder-print(root(T))`  
סיבוכיות זמן:  $\Theta(n)$  כאשר  $n$  הוא מספר הצמתים בעץ

# סרייה עץ בינארי



d h e b

## סרייה סופית postorder

סורך את תת העץ השמאלי

סורך את תת העץ ימני

בקר בשורש

הדפסת כל מפתחות העץ בסדר inorder

inorder-print( $x$ )

if  $x \neq \text{null}$  then

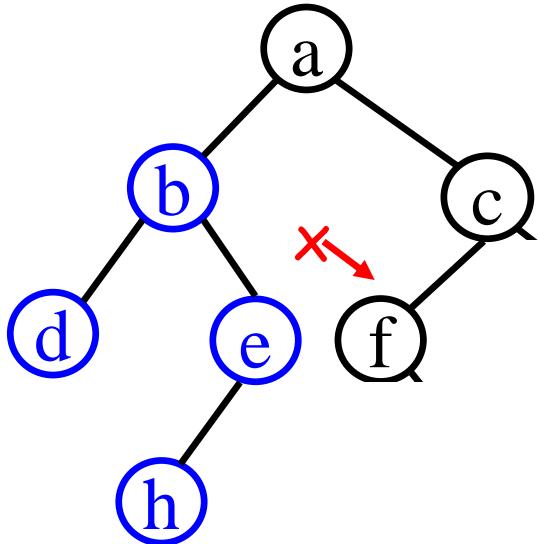
postorder-print(left( $x$ ))

postorder-print(right( $x$ ))

print key( $x$ )

קריאה:  $\text{postorder-print}(\text{root}(T))$   
סיבוכיות זמן:  $(n)\Theta$  כאשר  $\Theta$  הוא מספר הצלמים בעץ

# סרייה עץ בינארי



**סרייה סופית postorder**

סורך את תת העץ השמאלי

סורך את תת העץ הימני

בקר בשורש

הדפסת כל מפתחות העץ בסדר inorder

inorder-print( $x$ )

if  $x \neq \text{null}$  then

postorder-print(left( $x$ ))

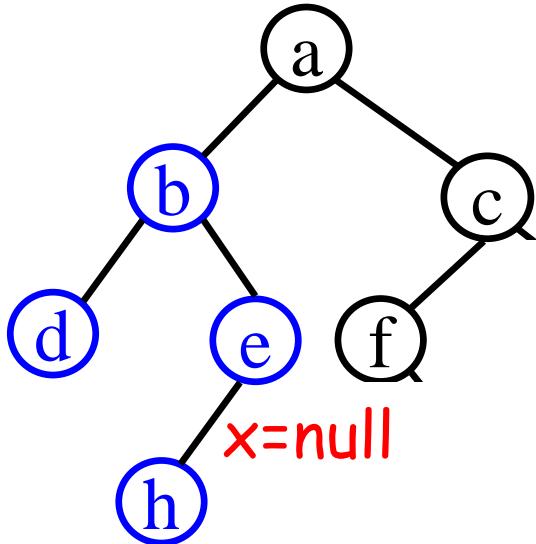
postorder-print(right( $x$ ))

print key( $x$ )

d h e b

קריאה: `postorder-print(root(T))`  
סיבוכיות זמן:  $\Theta(n)$  כאשר  $n$  הוא מספר הצמתים בעץ

# סרייה עץ בינארי



d h e b

**סרייה סופית postorder**

סורך את תת העץ השמאלי

סורך את תת העץ הימני

בקר בשורש

הדפסת כל מפתחות העץ בסדר inorder

inorder-print( $x$ )

if  $x \neq \text{null}$  then

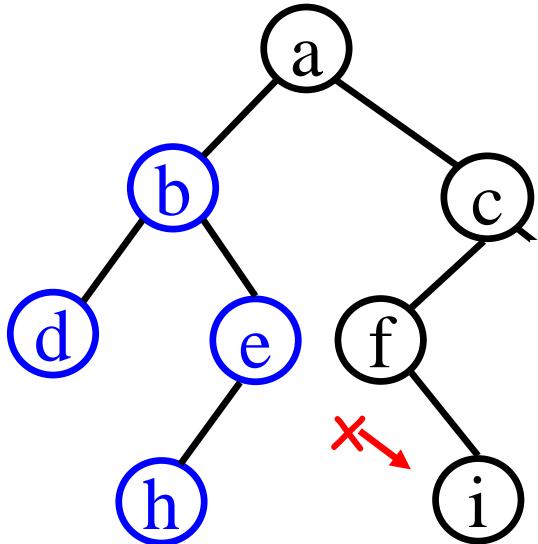
postorder-print(left( $x$ ))

postorder-print(right( $x$ ))

print key( $x$ )

קריאה: `postorder-print(root(T))`  
סיבוכיות זמן:  $\Theta(n)$  כאשר  $n$  הוא מספר הצמתים בעץ

# סרייה עץ בינארי



d h e b

## סדרה סופית postorder

סורך את תת העץ השמאלי

סורך את תת העץ ימני

בקר בשורש

הדפסת כל מפתחות העץ בסדר inorder

inorder-print( $x$ )

if  $x \neq \text{null}$  then

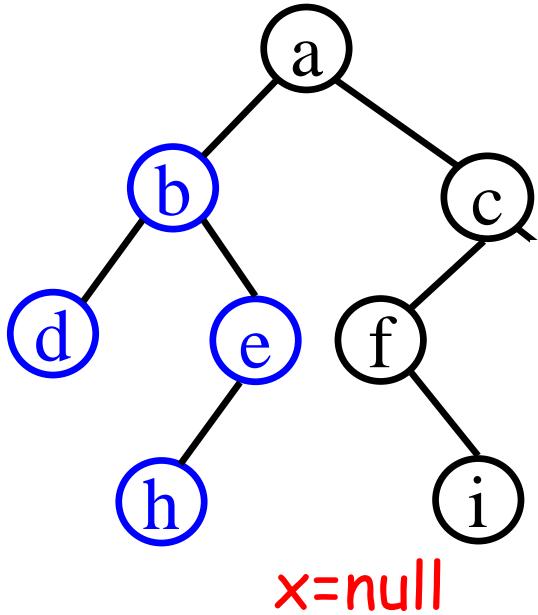
postorder-print(left( $x$ ))

postorder-print(right( $x$ ))

print key( $x$ )

קריאה: `postorder-print(root(T))`  
סיבוכיות זמן:  $\Theta(n)$  כאשר  $n$  הוא מספר הצמתים בעץ

# סרייה עץ בינארי



d h e b

**סרייה סופית postorder**

סורך את תת העץ השמאלי

סורך את תת העץ הימני

בקר בשורש

הדפסת כל מפתחות העץ בסדר inorder

inorder-print( $x$ )

if  $x \neq \text{null}$  then

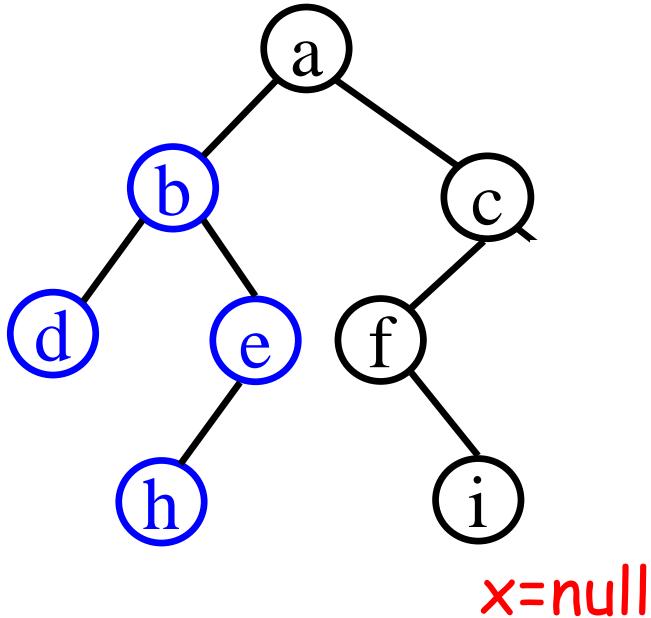
postorder-print(left( $x$ ))

postorder-print(right( $x$ ))

print key( $x$ )

קריאה: `postorder-print(root(T))`  
סיבוכיות זמן:  $\Theta(n)$  כאשר  $n$  הוא מספר הצמתים בעץ

# סרייה עץ בינארי



d h e b

## סרייה סופית postorder

סrox את תת העץ השמאלי

סrox את תת העץ ימני

בקר בשורש

הדפסת כל מפתחות העץ בסדר inorder

inorder-print( $x$ )

if  $x \neq \text{null}$  then

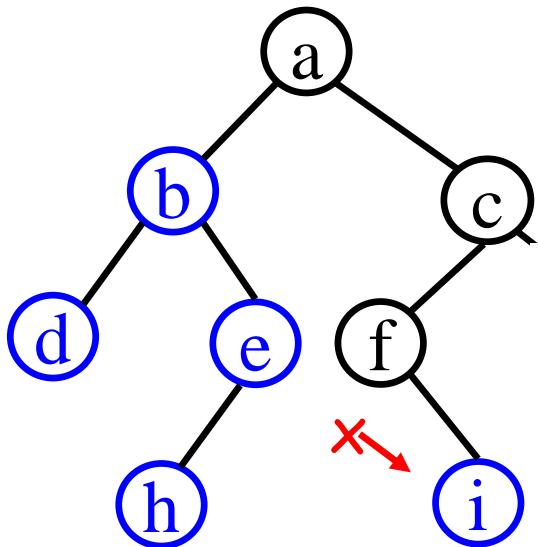
postorder-print(left( $x$ ))

postorder-print(right( $x$ ))

print key( $x$ )

קריאה: `postorder-print(root(T))`  
סיבוכיות זמן:  $\Theta(n)$  כאשר  $n$  הוא מספר הצמתים בעץ

# סרייה עץ בינארי



d h e b i

## סדרה סופית postorder

סורך את תת העץ השמאלי

סורך את תת העץ הימני

**בקר בשורש**

הדפסת כל מפתחות העץ בסדר inorder

inorder-print( $x$ )

if  $x \neq \text{null}$  then

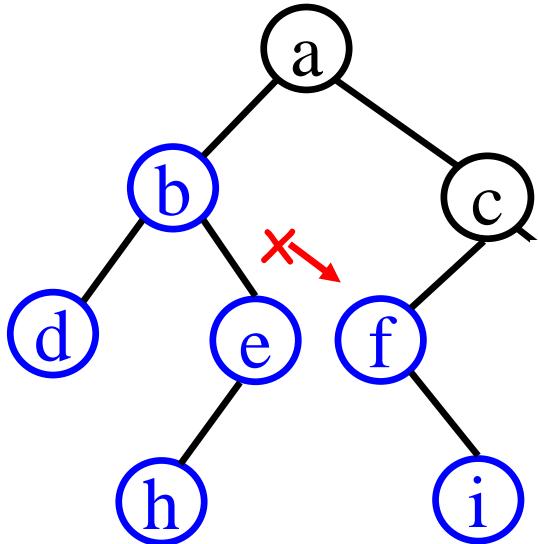
postorder-print(left( $x$ ))

postorder-print(right( $x$ ))

print key( $x$ )

קריאה:  $\text{postorder-print}(\text{root}(T))$   
סיבוכיות זמן:  $(n)\Theta$  כאשר  $\Theta$  הוא מספר הצמתים בעץ

# סרייה עץ בינארי



d h e b i f

## סדרה סופית postorder

סורך את תת העץ השמאלי

סורך את תת העץ הימני

**בקר בשורש**

הדפסת כל מפתחות העץ בסדר inorder

inorder-print( $x$ )

if  $x \neq \text{null}$  then

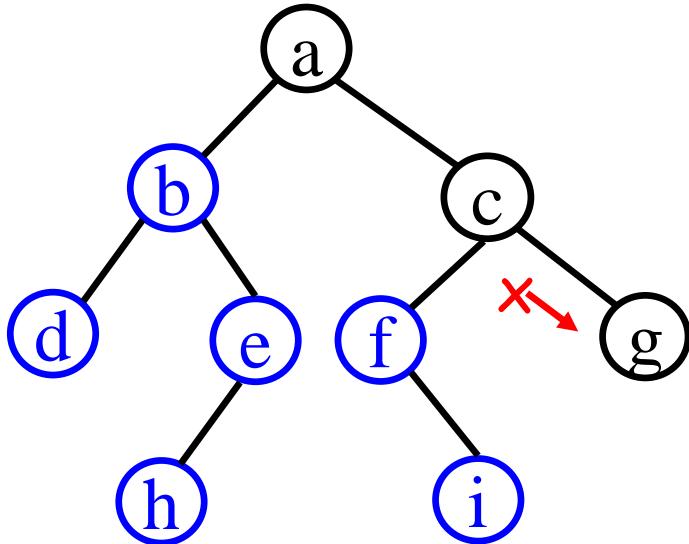
postorder-print(left( $x$ ))

postorder-print(right( $x$ ))

print key( $x$ )

קריאה: `postorder-print(root(T))`  
סיבוכיות זמן:  $\Theta(n)$  כאשר  $n$  הוא מספר הצמתים בעץ

# סרייה עץ בינארי



d h e b i f

## סדרה סופית postorder

סורך את תת העץ השמאלי

סורך את תת העץ הימני

בקר בשורש

הדפסת כל מפתחות העץ בסדר inorder

inorder-print( $x$ )

if  $x \neq \text{null}$  then

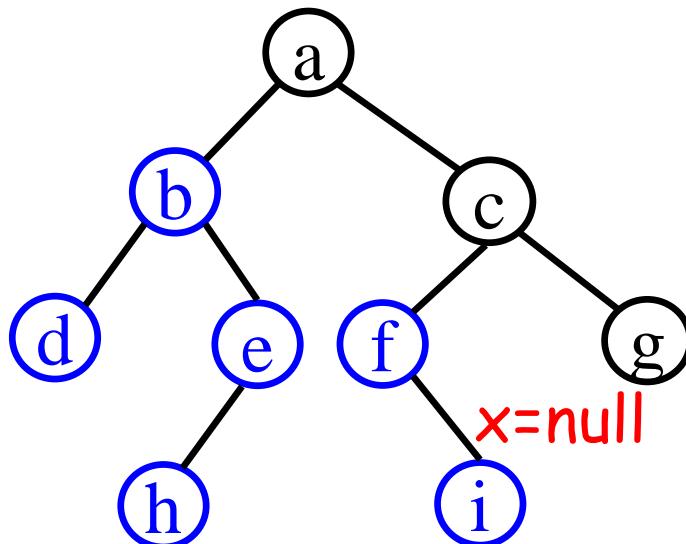
postorder-print(left( $x$ ))

postorder-print(right( $x$ ))

print key( $x$ )

קריאה: `postorder-print(root(T))`  
סיבוכיות זמן:  $(n)\Theta$  כאשר  $\Theta$  הוא מספר הצמתים בעץ

## סיקת עץ בינהי



d h e b i f

**inorder-print(x)**

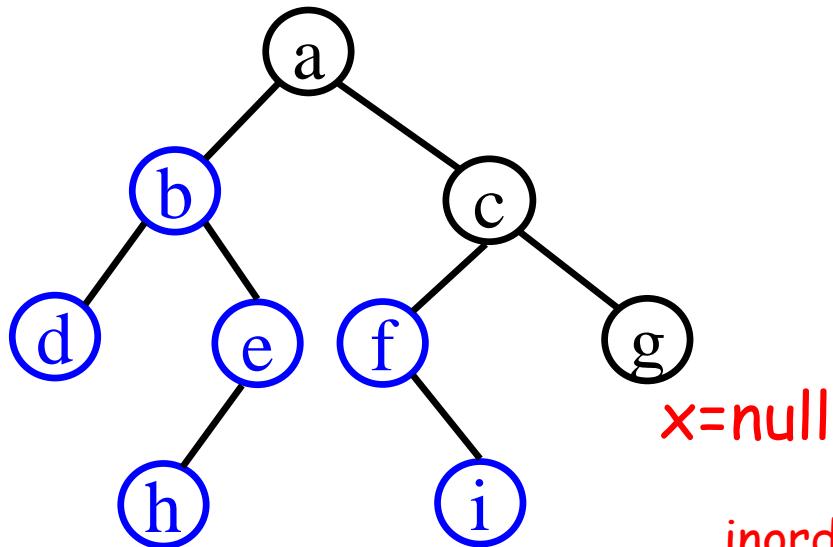
if  $x \neq \text{null}$  then

```
postorder-print(left(x))  
postorder-print(right(x))  
print key(x)
```

הקריאה: `postorder-print(root(T))`  
סיבוכיות זמן:  $\Theta(n)$  כאשר  $n$  הוא מספר הצלמתים בעץ

1

# סרייה עץ בינארי



d h e b i f

## סדרה סופית postorder

סורך את תת העץ השמאלי

סורך את תת העץ הימני

בקר בשורש

הדפסת כל מפתחות העץ בסדר inorder

inorder-print( $x$ )

if  $x \neq \text{null}$  then

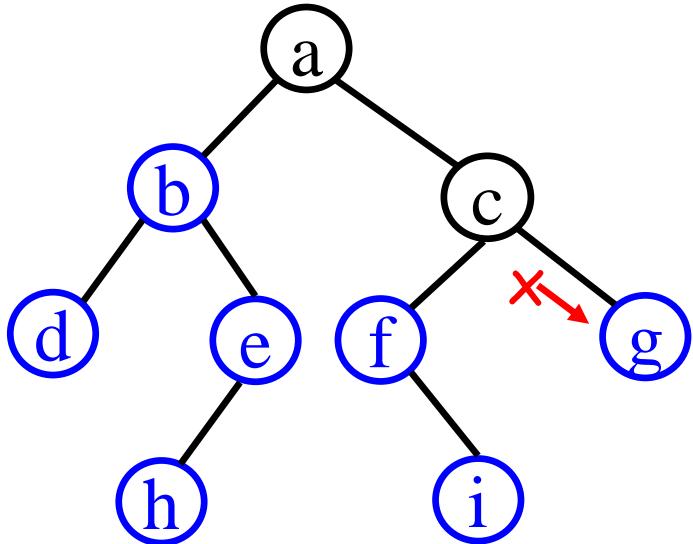
postorder-print(left( $x$ ))

postorder-print(right( $x$ ))

print key( $x$ )

קריאה: `postorder-print(root(T))`  
סיבוכיות זמן:  $\Theta(n)$  כאשר  $n$  הוא מספר הצמתים בעץ

# סרייה עץ בינארי



d h e b i f g

## סדרה סופית postorder

סורך את תת העץ השמאלי

סורך את תת העץ הימני

**בקר בשורש**

הדפסת כל מפתחות העץ בסדר inorder

inorder-print( $x$ )

if  $x \neq \text{null}$  then

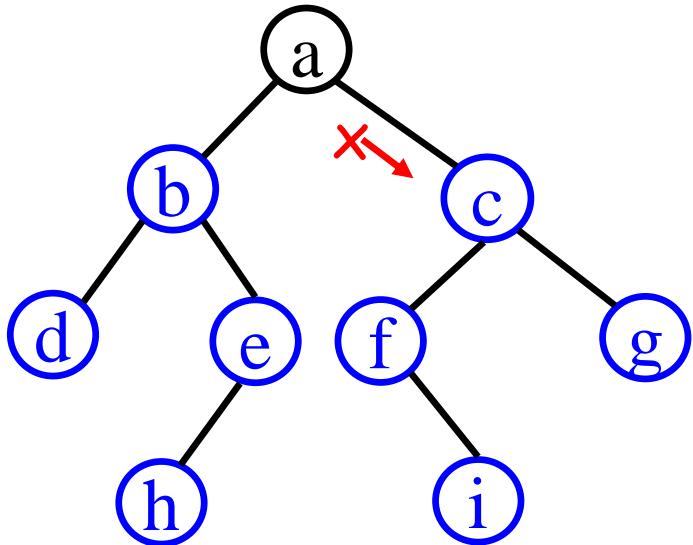
postorder-print(left( $x$ ))

postorder-print(right( $x$ ))

print key( $x$ )

קריאה: `postorder-print(root(T))`  
סיבוכיות זמן:  $(n)\Theta$  כאשר  $\Theta$  הוא מספר הצמתים בעץ

# סרייה עץ בינארי



d h e b i f g c

## סדרה סופית postorder

סורך את תת העץ השמאלי

סורך את תת העץ הימני

**בקיר בשורש**

הדפסת כל מפתחות העץ בסדר inorder

**inorder-print(x)**

**if**  $x \neq \text{null}$  **then**

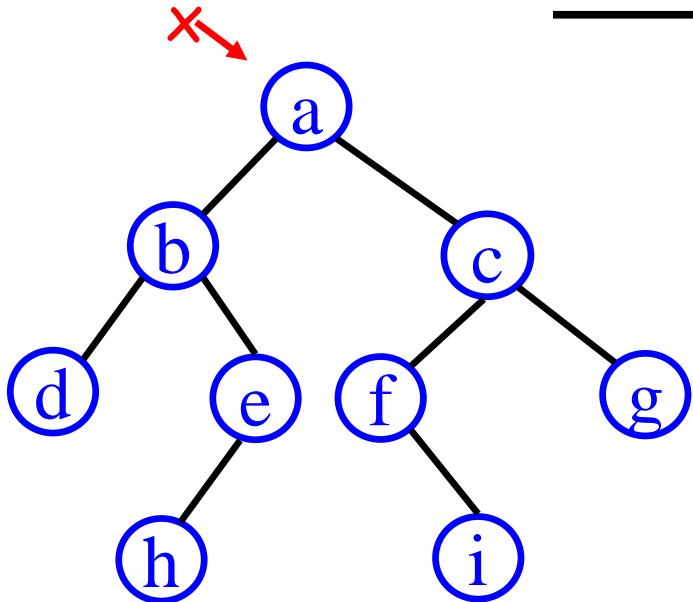
postorder-print(left( $x$ ))

postorder-print(right( $x$ ))

print key( $x$ )

קריאה: `postorder-print(root(T))`  
סיבוכיות זמן:  $(n)\Theta$  כאשר  $\Theta$  הוא מספר הצמתים בעץ

# סרייה עץ בינארי



d h e b i f g c a

## סרייה סופית postorder

סrox את תת העץ השמאלי

סrox את תת העץ הימני

**בקר בשורש**

הדפסת כל מפתחות העץ בסדר inorder

inorder-print( $x$ )

if  $x \neq \text{null}$  then

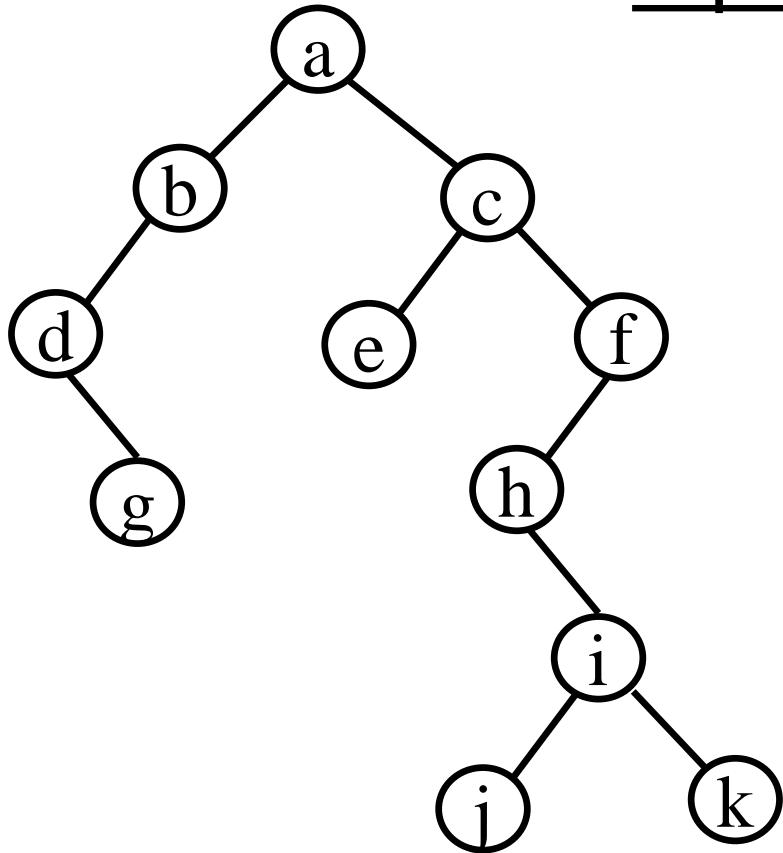
postorder-print(left( $x$ ))

postorder-print(right( $x$ ))

print key( $x$ )

קריאה: `postorder-print(root(T))`  
סיבוכיות זמן:  $(n)\Theta$  כאשר  $\Theta$  הוא מספר הצמתים בעץ

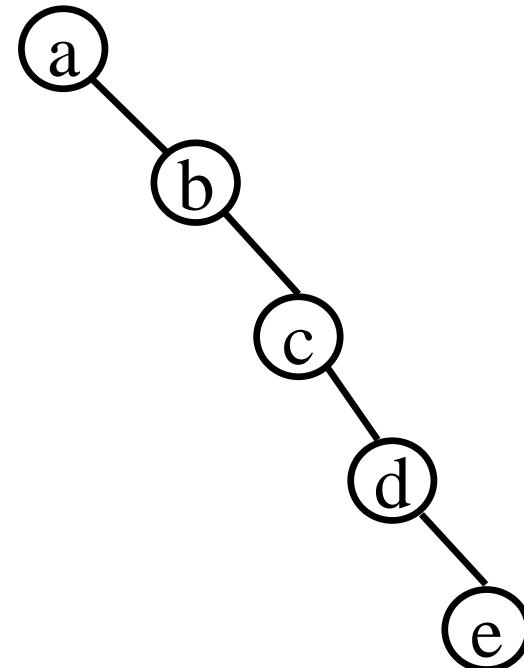
## דוגמאות לסריקות



preorder: a b d g c e f h i j k

inorder: d g b a e c h j i k f

postorder: g d b e j k i h f c a



preorder: a b c d e

inorder: a b c d e

postorder: e d c b a

## דוגמאות לסריקות

preorder: a b d g c e f h i j k

inorder: d g b a e c h j i k f

## דוגמאות לסריקות

שורש העץ

preorder: **a** b d g c e f h i j k

inorder: d g b a e c h j i k f

## דוגמאות לסריקות

שורש העץ

preorder: **a** b d g c e f h i j k

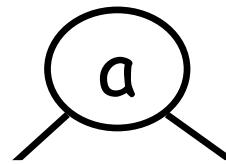
inorder: d g b **a** e c h j i k f

## דוגמאות לסריקות

שורש העץ

preorder: **a** b d g c e f h i j k

inorder:   d g b **a** e c h j i k f  
              \u2044        \u2044  
              תת העץ השמאלי      תת העץ הימני



d g b      e c h j i k f

## דוגמאות לסריקות

שורש העץ

preorder: **a** b d g c e f h i j k

inorder:   d g b **a** e c h j i k f  
              \u2044        \u2044  
              תת העץ השמאלי      תת העץ הימני

## דוגמאות לסריקות

preorder: b d g

inorder: d g b  
  
תת העץ השמאלי

## דוגמאות לסריקות

שורש העץ

preorder:    b d g

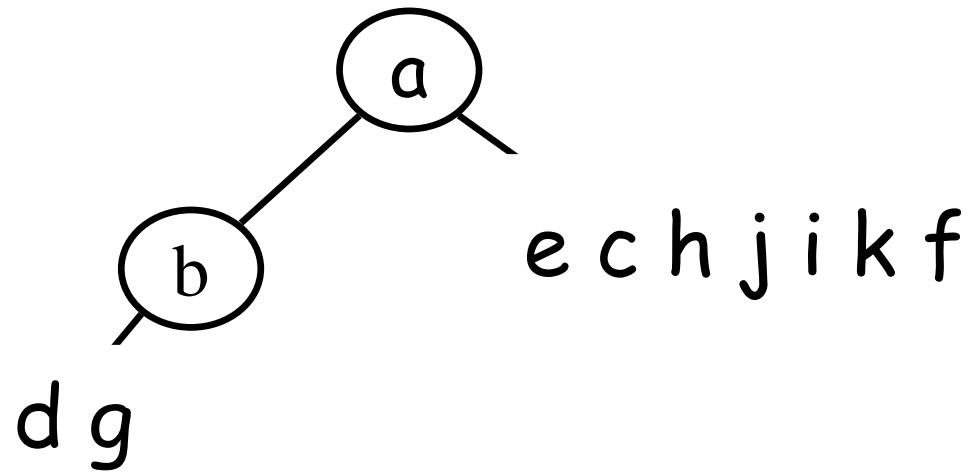
inorder:    d g b

## דוגמאות לסריקות

שורש העץ

preorder: b d g

inorder: d g **b**  
        └── └──  
        תת העץ ימני   תת העץ השמאלי



## דוגמאות לסריקות

שורש העץ

preorder: b d g

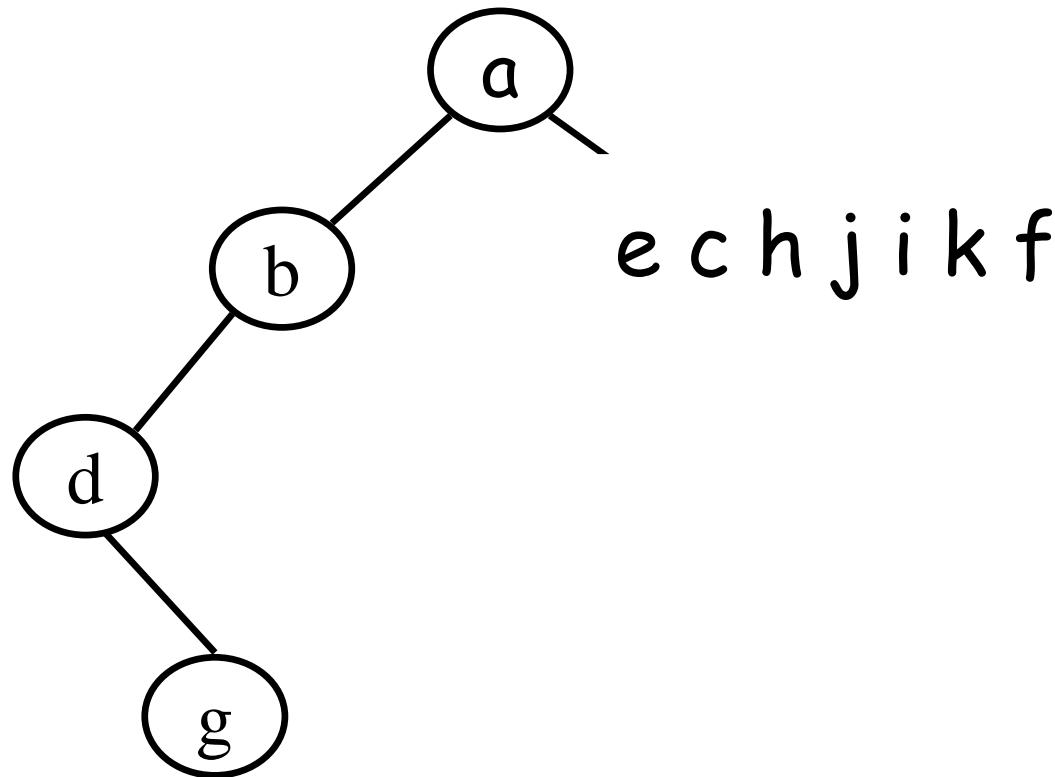
inorder: d g **b**  
        └── └──  
        תת העץ ימני   תת העץ השמאלי

## דוגמאות לסריקות

שורש העץ

preorder: d g

inorder: d g  
             ב' ב'  
תת העץ הימני    תת העץ השמאלי



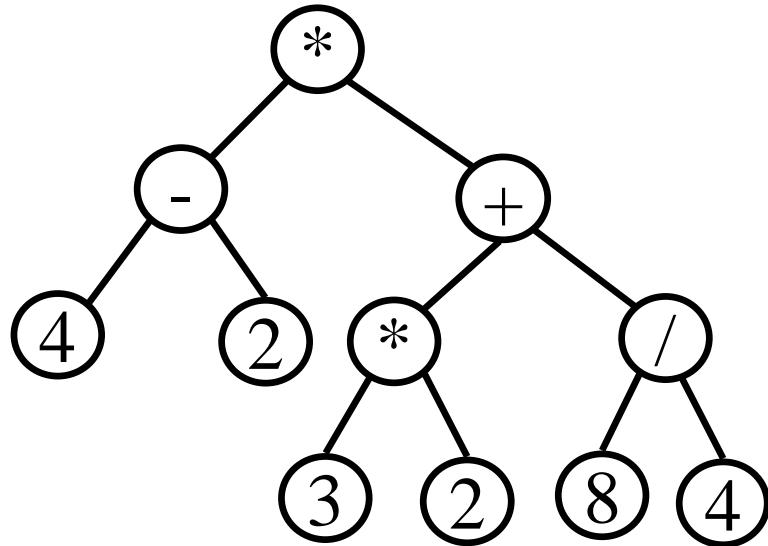
## דוגמאות לסריקות

שורש העץ

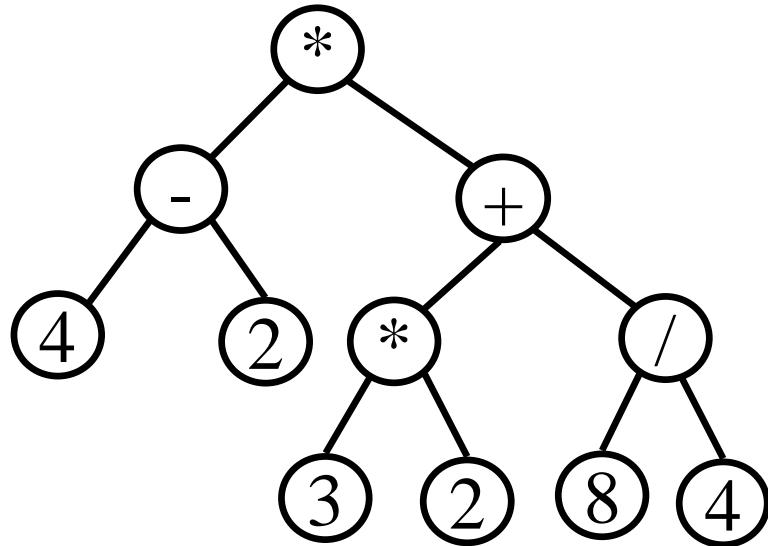
preorder: **a** b d g c e f h i j k

inorder:   d g b **a** e c h j i k f  
              \u2044        \u2044  
              תת העץ השמאלי      תת העץ הימני

## דוגמה: עץ המיצג ביטוי ארכיטמטי

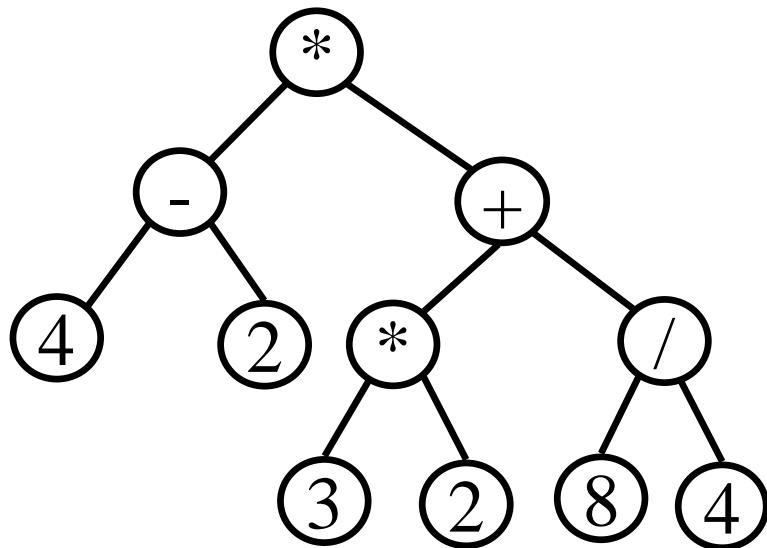


## דוגמה: עץ המיצג ביטוי ארכיטמטי



$$4 - 2 * 3 * 2 + 8 / 4$$

# דוגמה: עץ המיצג ביטוי ארכיטמטי



הדפסת הביטוי ב-*infix*

*inorder-print(x)*

*if*  $x \neq \text{null}$  *then*

*inorder-print(left(x))*

*print key(x)*

*inorder-print(right(x))*

$4 - 2 * 3 * 2 + 8 / 4$

קריאה: *inorder-print(root(T))*

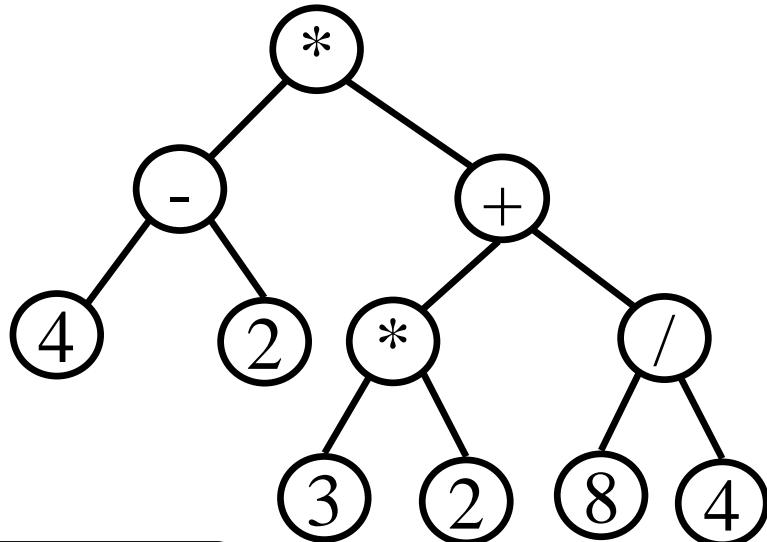
סירה תומכת *inorder*

סרוק את תת העץ השמאלי

בקר בשורש

סרוק את תת העץ הימני

# דוגמה: עץ המיצג ביטוי ארכיטמטי



$4 - 2 * 3 * 2 + 8 / 4$

סרייה תומכת inorder  
סורך את תת העץ השמאלי  
בקר בשורש  
סורך את תת העץ הימני

הדפסת הביטוי ב-prefix

**inorder-print(x)**

**if**  $x \neq \text{null}$  **then**

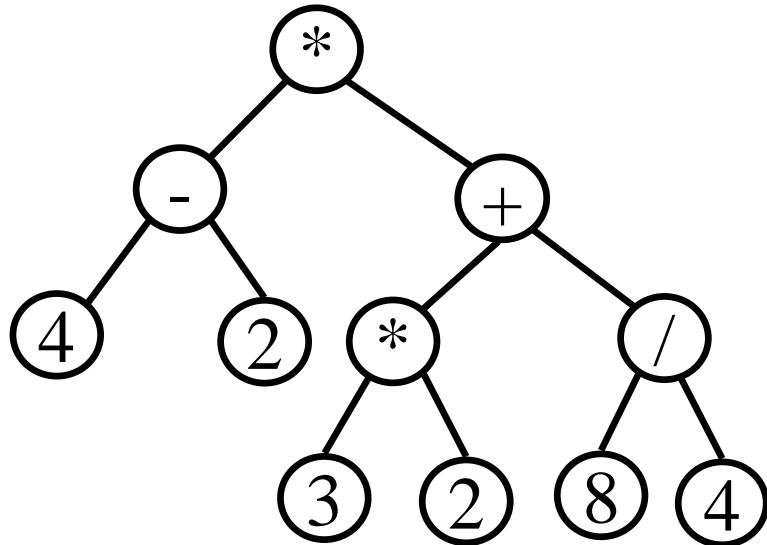
inorder-print(left( $x$ ))

print key( $x$ )

inorder-print(right( $x$ ))

הקריאה: inorder-print(root(T))

# דוגמה: עץ המיצג ביטוי ארכיטמטי



סרייה תומכת inorder

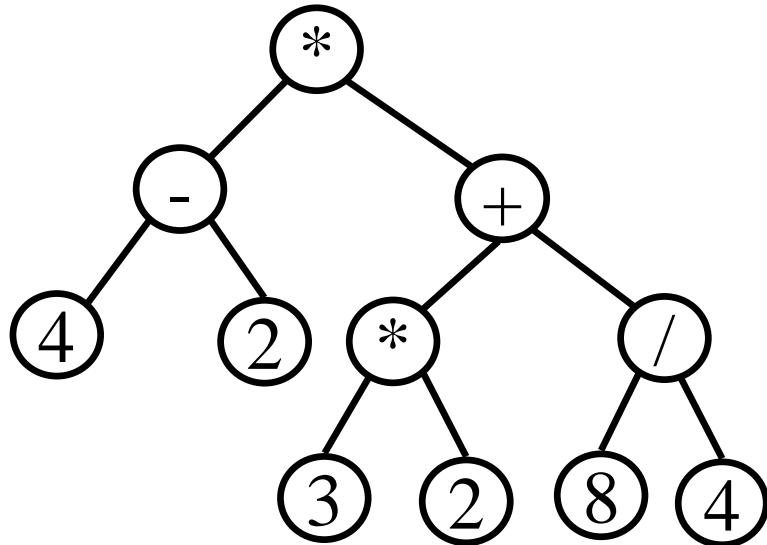
סרוק את תת העץ השמאלי

בקר בשורש

סרוק את תת העץ הימני

$$(((4)-(2))*(((3)*(2))+((8)/(4))))$$

# דוגמה: עץ המיצג ביטוי ארכיטמטי



$((4)-(2)) * (((3)*(2)) + ((8)/(4))))$

סרייה תומכת inorder

סרוק את תת העץ השמאלי

בקר בשורש

סרוק את תת העץ הימני

הדפסת הביטוי ב-x

**inorder-print(x)**

**if**  $x \neq \text{null}$  **then**

print("(")

inorder-print(left( $x$ ))

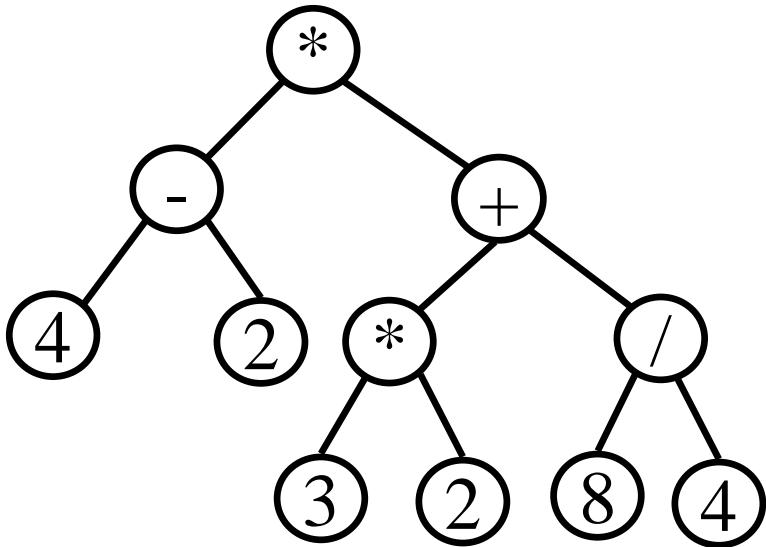
print key( $x$ )

inorder-print(right( $x$ ))

print(")")

קריאה: `inorder-print(root(T))`

# דוגמה: עץ המיצג ביטוי ארכיטמטי



סרייה תומכת inorder

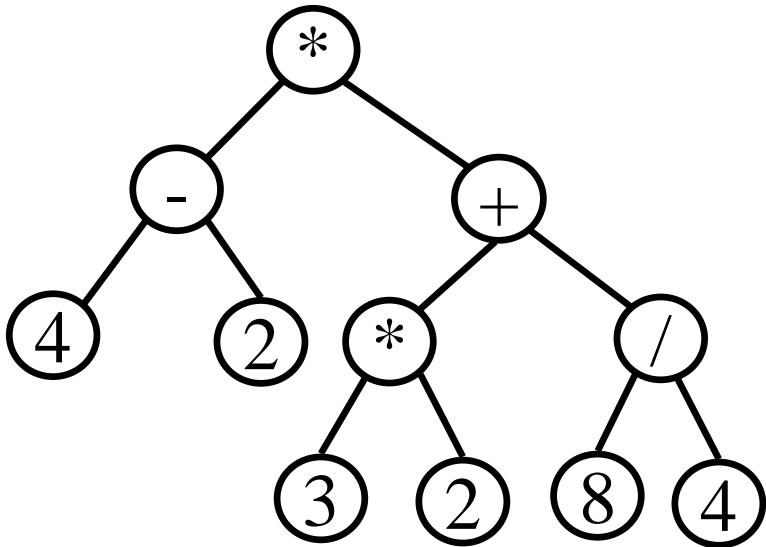
סרוק את תת העץ השמאלי

בקר בשורש

סרוק את תת העץ הימני

$$((4-2)*((3*2)+(8/4)))$$

# דוגמה: עץ המיצג ביטוי ארכיטמטי



$$((4-2)*((3*2)+(8/4)))$$

הדפסת הביטוי ב-*infix*

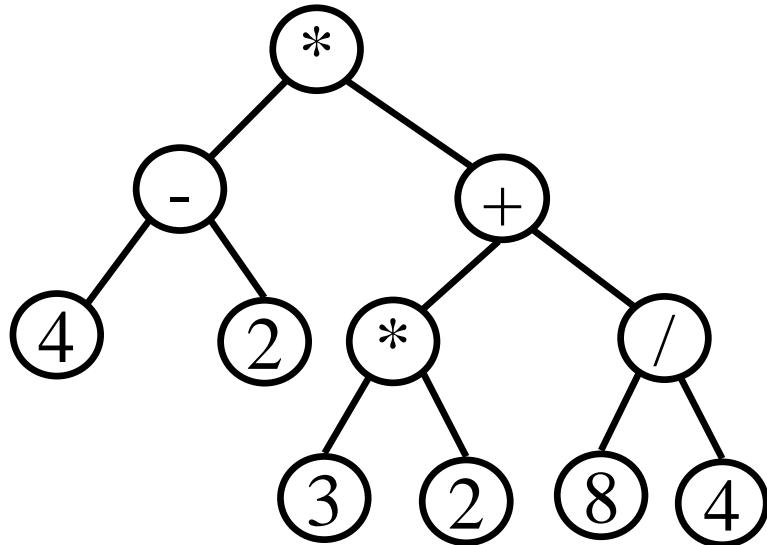
*inorder-print(x)*  
if  $x \neq \text{null}$  then

```

if  $\text{left}(x) \neq \text{null}$  or  $\text{right}(x) \neq \text{null}$  print("(")
inorder-print(left(x))
print key(x)
inorder-print(right(x))
if  $\text{left}(x) \neq \text{null}$  and  $\text{right}(x) \neq \text{null}$  print(")")
  
```

הקריאה: *inorder-print(root(T))*

# דוגמה: עץ עבר ביטוי ארכטמטי



סריקה סופית postorder

סרק את תת העץ השמאלי  
סרק את תת העץ הימני  
בקיר בשורש

הדפסת הביטוי ב postfix

**postorder-print(x)**

**if**  $x \neq \text{null}$  **then**

postorder-print(left( $x$ ))

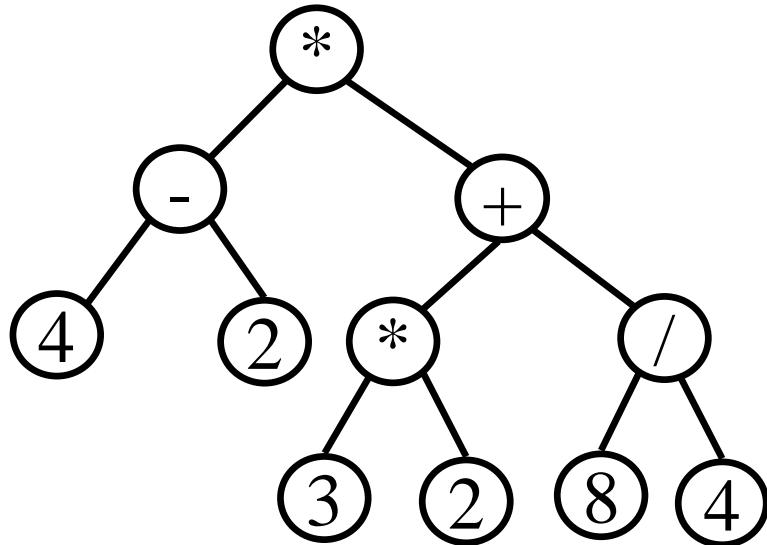
postorder-print(right( $x$ ))

print key( $x$ )

קריאה:

postorder-print(root( $T$ ))

# דוגמה: עץ עבר ביטוי ארכטמי



$42 - 32 * 84 / + *$



4-2

סיריקה סופית postorder

סrox את תת העץ השמאלי  
סrox את תת העץ הימני  
בקר בשורש

הדפסת הביטוי ב postfix

postorder-print( $x$ )

if  $x \neq \text{null}$  then

postorder-print(left( $x$ ))

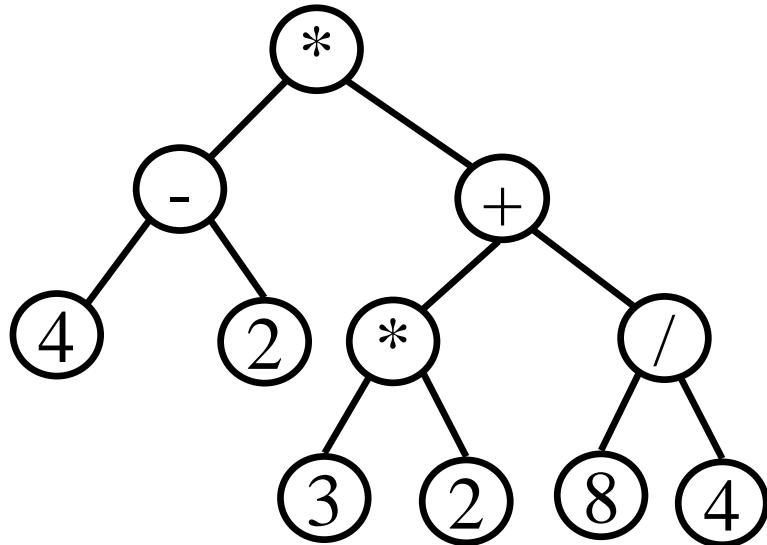
postorder-print(right( $x$ ))

print key( $x$ )

קריאה:

postorder-print(root( $T$ ))

# דוגמה: עץ עבר ביטוי ארכטמי



$42 - 32 * 84 / + *$



$4-2 \quad 3*2$

סיריקה סופית postorder

סrox את תת העץ השמאלי  
סrox את תת העץ הימני  
בקר בשורש

הדפסת הביטוי ב postfix

**postorder-print(x)**

**if**  $x \neq \text{null}$  **then**

postorder-print(left( $x$ ))

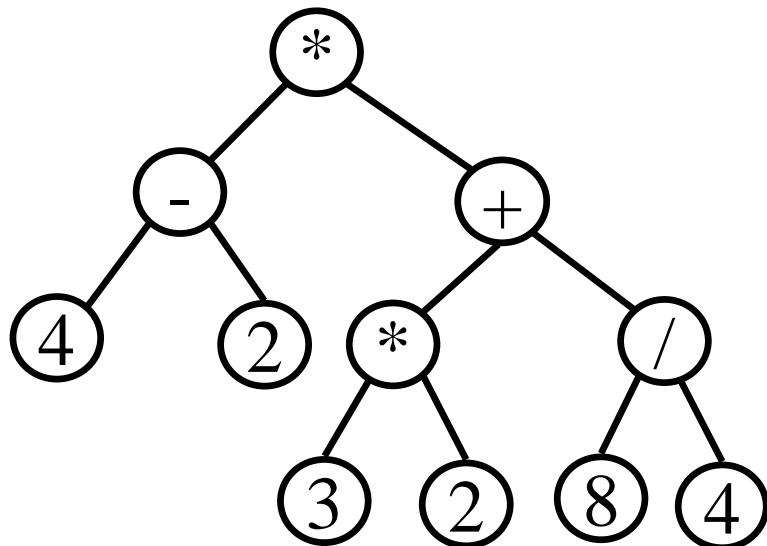
postorder-print(right( $x$ ))

print key( $x$ )

קריאה:

postorder-print(root( $T$ ))

# דוגמה: עץ עבר ביטוי ארכטמטי



$42 - 32 * 84 / + *$



$4 - 2$     $3 * 2$     $8 / 4$

סיריקה סופית postorder

סrox את תת העץ השמאלי  
סrox את תת העץ הימני  
בקר בשורש

הדפסת הביטוי ב postfix

**postorder-print(x)**

**if**  $x \neq \text{null}$  **then**

postorder-print(left( $x$ ))

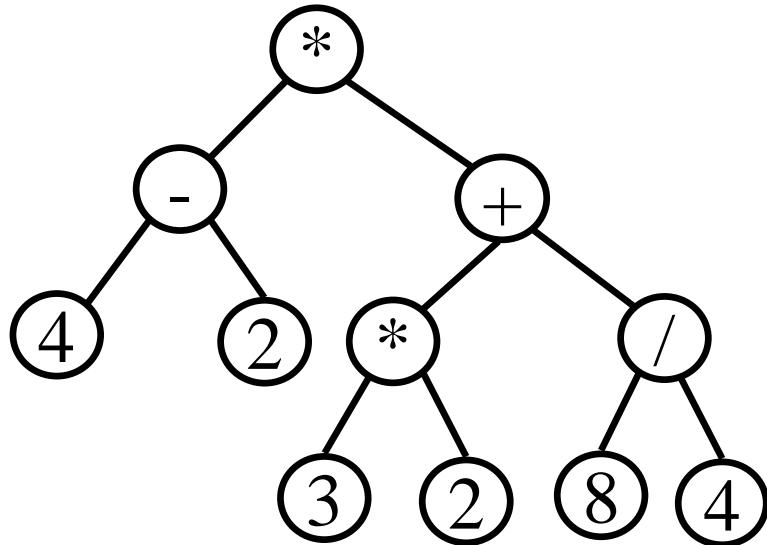
postorder-print(right( $x$ ))

print key( $x$ )

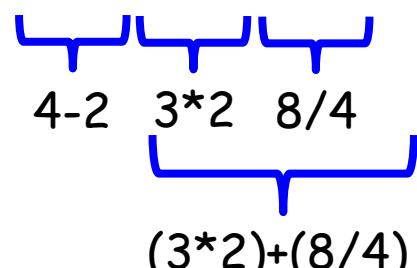
קריאה:

postorder-print(root( $T$ ))

# דוגמה: עץ עבר ביטוי ארכטמי



$42 - 32 * 8 / 4 + *$



סיריקה סופית postorder

סורך את תת העץ השמאלי  
סורך את תת העץ הימני  
בקיר בשורש

הדפסת הביטוי ב postfix

postorder-print( $x$ )

if  $x \neq \text{null}$  then

postorder-print(left( $x$ ))

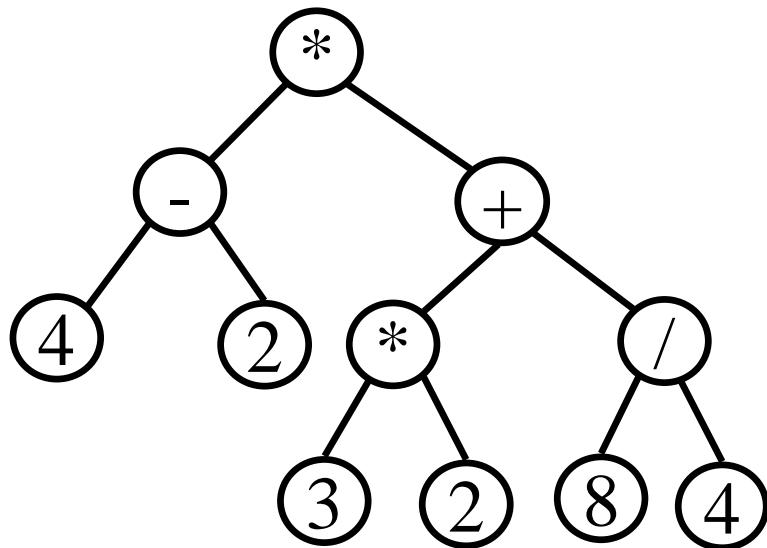
postorder-print(right( $x$ ))

print key( $x$ )

קריאה:

postorder-print(root( $T$ ))

# דוגמה: עץ עבר ביטוי ארכטמי



$42 - 32 * 8 / 4$

$\overbrace{\quad}^4 - \overbrace{\quad}^{3*2} \overbrace{\quad}^{8/4}$

$4 - 2$

$(3*2)$

$(3*2) + (8/4)$

$(4-2)*((3*2)+(8/4))$

סירקה סופית postorder

סורך את תת העץ השמאלי  
סורך את תת העץ הימני  
בקיר בשורש

הדפסת הביטוי ב postfix

postorder-print( $x$ )

if  $x \neq \text{null}$  then

postorder-print(left( $x$ ))

postorder-print(right( $x$ ))

print key( $x$ )

קריאה:

postorder-print(root( $T$ ))