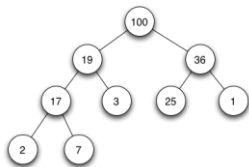


## תזכורת: Heaps

- עץ בינארי מלא
- החוק הבסיסי
  - אם צומת B צאצא של צומת A אזי  $Key(A) \leq Key(B)$
- הפעולות הנתמכות
  - Find-min
  - Delete-min
  - Decrease-key
  - Insert
  - Merge



## מבני נתונים 08a

תרגול 8  
7.2.2008

## ערמות



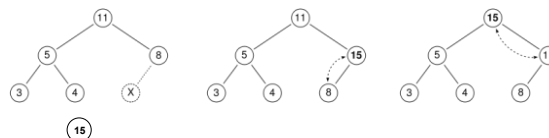
ליאור שפירא

## תרגיל 1

- בהינתן מערך באורך n, נרצה ליצור min heap ע"י הכנסה סדרתית של ערכי המערך. הראו סדרת הפעולות לוקחת  $\Omega(n \log n)$  במקרה הכי גרוע (worst case)

## תזכורת: Heaps

הוספת צומת (עבור max-heap)



מחיקת השורש



## תרגיל 3 – Median Heap

- ממשו מבנה נתונים התומך בפעולות
  - insert בזמן  $O(\log n)$
  - extract-median בזמן  $O(\log n)$
  - find-median בזמן  $O(1)$

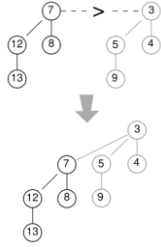


## תרגיל 2

- בהינתן heap שתומך בפעולות extract-min ו-insert בזמן amortized  $f(n)$ , הראו שניתן למיין מערך מגודל n בזמן  $O(n \cdot f(n))$

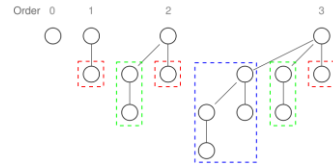
## תזכורת: Binomial Heaps

- איחוד שני עצים בינומיים מסדר  $k-1$
- ניצור עץ בינומי מסדר  $k$  ע"י תליית אחד העצים כבן השמאלי ביותר של העץ השני



## תזכורת: Binomial Heaps

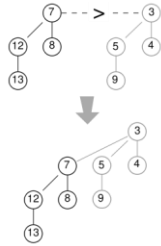
- תחילה נגדיר Binomial Tree:
  - עץ בינומי מדרגה 0 מכיל צומת יחידה
  - עץ בינומי מדרגה  $k$  הוא
    - בעל שורש מדרגה  $k$
    - ילדיו הינם עצים בינומיים מדרגות  $0, k-2, \dots, k-1$  (בסדר זה)
    - עץ בינומי מדרגה  $k$  מכיל  $2^k$  צמתים והינו בגובה  $k$



## תזכורת: Binomial Heaps

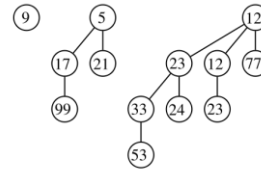
- פעולת Merge של שני עצים מדרגה  $k$

```
function mergeTree(p, q)
  if p.root <= q.root
    return p.addSubTree(q)
  else
    return q.addSubTree(p)
end
```



## תזכורת: Binomial Heaps

- הגדרת Binomial Heap
  - סט עצים בינומיים המקיימים את התכונות הבאות
    - כל עץ מקיים את תכונת minimum-heap (כל יצאצא גדול מהורה שלו)
    - עבור כל סדר  $k$  של עץ בינומי, יש 0 או 1 עצים כאלו ב-heap



## תזכורת: Binomial Heaps

- פעולת Insert
  - ניצור heap חדש המכיל את האבר החדש, ונבצע merge בין שני ה-heaps
- פעולת minimum
  - עלינו לחפש את הערך המינימלי מבין שורשי העצים בheap
- פעולת delete-min
  - מצא את האבר ונחק אותו
  - הפוך את בניו ל-binomial heap ומג את שני ה-heaps
- פעולת Decrease-min
  - בדומה לפעולות ב-heap רגיל
- פעולת Delete
  - שנה את הערך ל-∞ ובצע delete-min

## תזכורת: Binomial Heaps

- פעולת merge של שני Binomial Heaps

```
function merge(p, q)
  while not ( p.end() and q.end() )
    tree = mergeTree(p.currentTree(), q.currentTree())
    if not heap.currentTree().empty()
      tree = mergeTree(tree, heap.currentTree())
      heap.addTree(tree)
    else
      heap.addTree(tree)
    end if
    heap.next() p.next() q.next()
  end while
end
```

הדגמה באימיציה של Binomial Heaps: <http://www.cse.yorku.ca/~aaw/Solutions/BinomialHeap.html>

## תרגיל: עצים בינומיים שמנים

□ כיצד נגדיר ערמה בינומית "שמנה"?

ערמה בינומית שמנה



□ כיצד תתבצע פעולת meld? למה הדבר אנלוגי?

## תרגיל: עצים בינומיים שמנים

□ נגדיר עצים בינומיים שמנים כך:

- עץ בינומי "שמן" מדרגה 0 מכיל צומת אחד בלבד.
- עץ בינומי "שמן" מדרגה  $k$  ניתן לבנות משלושה עצים בינומיים "שמנים" מדרגה  $k-1$  כאשר נתלה שניים מהם על העץ השלישי.

