

תרגיל 11 לעבודה עצמית – עץ ערמה HEAP

שאלה 1

נתון מערך של מספרים שלמים: 12, 19, 10, 4, 23, 7, 45, 8, 15. סרטט Max-Heap עבור המספרים האלה.

שאלה 2 סרטט Min-Heap עבור המספרים של שאלה 1.

שאלה 3

- (א) כמה עליים יש לעץ ערמה?
- (ב) מהו האינדקס של העלה הראשון?
- (ג) האם המערך שבו נשמר עץ ערמה הוא חייב להיות ממוין?

שאלה 4

נתון עץ ערמה כלשהו. נתון ערך של הצומת שצריך לעדכן (increase or decrease) אותו. איך אפשר למצוא את הצומת שמכיל את הערך הנתון? הוסף לעץ ערמה פונקציה שמעדכנת את הערך הנתון. מהי הסיבוכיות של אלגוריתם זה?

שאלה 5

נתון Max-Heap שמכיל מספרים שלמים ומספר שלם K . הוסף לעץ ערמה פונקציה שמדפיסה את כל איברי של Max-Heap הגדולים מ- K .

שאלה 6

כתוב פונקציה שמקבלת מערך של מספרים שלמים. הפונקציה מחזירה true אם המערך מהווה Max-Heap, אחרת היא מחזירה false.

שאלה 7

ממש את Min-Heap, השומרת על תכונה של Min-Heap: $A[\text{Parent}(i)] \leq A[i]$ (ערכו השורש קטן או שווה לערכי הבנים שלו).

ממש את המתודות:

- בנאי המחלקה, שמקבל מערך של שלמים ומעתיק אתו כמשתנה עצם המחלקה

public HeapMinImplementation(int arr[])

- **,private int parent(int i)**

מתודה שמחזירה את אינדקס של קדקוד האב של קדקוד שאינדקס הוא i .

- **private int leftChild(int p)**

מתודה שמחזירה את אינדקס של בן השמאלי של קדקוד p .

- **private int rightChild(int p)**

מתודה שמחזירה את אינדקס של בן הימני של קדקוד p .

- **,private void minHeapify(int v, int heapSize)**

מתודה ששומרת על תכונה של Min-Heap.

- **– public void buildMinHeap()**

מתודה הבונה Min-Heap ממערך נתון.

- **public void heapSort()**

מתודה הממיינת מערך בסדר יורד בעזרת Min-Heap.

- **public int heapExtractMin()**

מתודה שמחזירה ומוחקת את שורש הערמה.

- **,private void heapDecreaseKey(int i, int key)**

מתודה המקטינה את ערכו של קדקוד i לערך חדש key .

- **public void minHeapInsert(int key)**

מתודה המוסיפה ערך חדש לערמה.