# בניית עץ מרשימת דרגות

**בעיה**: אם יש רשימת הדרגות של הקודוקודים בגרף, האם רשימה זאת מייצגת גרף.

**שאלה ספציפית**: האם גרף זה הוא עץ.

**דוגמא:**

האם רשימת הדרגות הבאה יכולה להיות של עץ?

1,1,1,2,2,3,4

**משפט:** סדרת מספרים כאשר מספר שלם חיובי, מייצגת דרגות הקודקודים בעץ אם ורק אם מתקיים את השוויון הבא:

. **,** כאשר

(\*)

**הוכחה:**

**הכרחיות**: אם T הוא עץ, אז לפי הגדרת העץ, בכל עץ .

ולפי למת לחיצת הידיים, סכום דרגות הקדקודים שווה לפעמיים מספר צלעות:

. לכן ,

אם כך, צריך לוודא שסכום הדרגות יהיה פעמיים מספר הצלעות.

אבל האם זה מספיק?

נתבונן ב**דוגמה 1**:

נסכום את האיברים

נסיק מסקנה לגבי כמות הקודקודים

נקבל סתירה כי לא יכול להיות חצי קודקוד.

נתבונן ב**דוגמה 2**:

נסכום את האיברים

נסיק מסקנה לגבי כמות הקודקודים

נקבל סתירה כי יש לנו ברשימה נתונים רק ל 7 קודקודים.

מתי רשימת דרגות היא יכולה להיות של עץ?

כאשר מתקיים

וגם יש לנו מספיק נתונים לכמות הקודקודים.

נתבונן ב**דוגמה 3**:

נסכום את האיברים

נסיק מסקנה לגבי כמות הקודקודים

מתקיים!

**מספקיות**: נתונה סדרת מספרים כאשר

המקיימת את השוויון .

צריך להוכיח כי הסדרה מייצגת את קודקודים העץ עם הדרגות הללו.

*הוכחה* באינדוקציה לפי מספר הקדקודים:



- לעץ יש רק קדקוד אחד ודרגתו שווה 0 והתנאי מתקיים.

התנאי מתקיים



בסיס: n=1 אז

ניקח לדוגמה גם n=2

הנחת אינדוקציה: הטענה נכונה עבור כלשהו.

שלב האינדוקציה: נוכיח את הטענה עבור n+1, כלומר

נמיין את מערך הדרגות בסדר יורד, מהגדול לקטן:

ידוע שלכל עץ יש לפחות שני עלים. אז בטוח: ו-.

בונים סדרה חדשה בגודל n ע"י מחיקת עלה וגם מקטינים ב-1 דרגה ל- :

, והסדרה החדשה מקיימת את התנאי:

ולפי הנחת האינדוקציה ניתן לבנות עץ שדרגתם הם .

עכשיו נבנה עץ בעל n+1 קדקודים: נחזיר לקדקוד 1 את העלה (קדקוד מספר n+1 ), דרגה של קודקוד 1 יגדל ב-1 וסכום הדרגות יהיה:

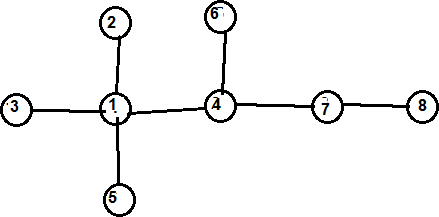
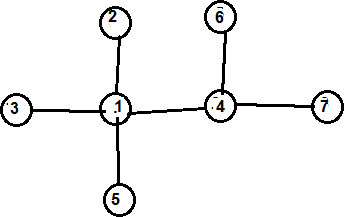
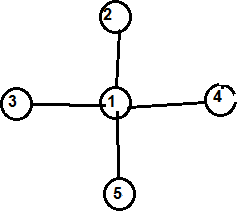
▄

# רעיון ליצירת העץ לפי הדרגות הנתונות שמקיימות את התנאי (\*)

נמיין את מערך הדרגות בסדר יורד, מהגדול לקטן:

בונים כוכב, שדרגת מרכז הכוכב הוא , ניקח אצל הכוכב עלה אחד ונהפוך אותו למרכז הכוכב בדרגה , ניקח עלה אחד ונהפוך אותו למרכז הכוכב בדרגה ונחזור לפעולה זו עד של לא נבנה את כל קדקודי העץ.

דוגמא: יהי n=8 סדרת הדרגות 4,3,2,1,1,1,1,1 המקיימת את התנאי (\*).



# (a) ( b) (c)

# Tree building pseudo – code

*n*

**Input**: Array of degrees d [n]: *di*  2(*n* 1), n – number of vertices. d[] = Sort(d[])

1

**Output**: tree – graph as adjacency list

# public static ArrayList<Integer>[] treeBuilding(int deg[]){ int n = deg.length;

ArrayList<Integer>[] tree = **new** ArrayList[n]; **for** (**int** i = 0; i < n; i++) {

tree[i] = **new** ArrayList<Integer>();

}

**int** first\_1 = 0; **while**(deg[first\_1]>1) first\_1++; **int** vertex = 0, numV = 1;

**if** (n >= 3){

**while**(vertex < first\_1) {

**for** (**int** j = 0; j < deg[vertex]; j++) { tree[vertex].add(numV); tree[numV].add(vertex);

numV++;

}

vertex++; deg[vertex]--;

}

}

**else if** (n==2){

tree[0].add(1);

tree[1].add(0);

}

**return** tree;

}

נראה הרצת האלגוריתם למספר הקודקודים n=8 וסדרת הדרגות: 4,3,2,1,1,1,1,1 שמקיימת את

התנאי (\*).

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |
| 1 | 1 | 1 | 1 | 1 | 2 | 3 | 4 |

first\_1 = 3

while(vertex < first\_1)

1. vertex = 0, numV = 1,

numV = 5

2

1

0

4

3

1. vertex = 1, numV = 5,

numV = 7

6

2

5

1

0

4

3

1. vertex = 2, numV = 7,

numV = 8

7

6

2

5

1

0

4

3