# **פתרון תרגיל 11**

## **מיון**

1) מהו זמן הריצה של מיון מהיר על מערך שכל איבריו זהים?

פתרון

בריצת partition שני האינדקסים יקודמו ב-1 ואז תבוצע החלפה. פעולה זו תחזור על עצמה עד ששני האינדקסים ייפגשו, וזה יקרה בדיוק באמצע המערך.

כך ייצא שחלוקת המערך תבוצע בדיוק באמצע (כלומר – הקריאות הרקורסיביות הבאות יקבלו מערכים באורך מחצית מהמערך שבשלב הקודם), ולכן זמן הריצה יהיה $T(n)=2T(\frac{n}{2})+\Theta(n)=\Theta(n\log n)$.

2) בהנתן רשימה של n איברים, תארו אלגוריתם אשר מחזיר את המספר שמופיע הכי הרבה פעמים ברשימה, ונתחו את הסיבוכיות שלו.

פתרון

שלב א – נמיין את הרשימה, לקבלת מערך ממוין A.

שלב ב – נרוץ על המערך הממוין למציאת האיבר השכיח ביותר על ידי הפונקציה הבאה, תוך שימוש במשתנים הבאים:

counter – מונה כמה מופעים יש מכל מספר (נקדם אותו כל עוד המספר מופיע כמה פעמים רצוף).

max – מקסימום המופעים שמצאנו בינתיים (של המספר השכיח ביותר).

common – האיבר השכיח ביותר שמצאנו עד עתה.

```
int max=0, counter =0;
int common;
for (i=2; i<=n ; i++)
    {  if A[i] ==A[i-1]
            counter ++;
       else
          { if ac > max
                common = A[i-1];
            counter=0
          }
```

```
}
// בדיקה עבור ה-counter האחרון שחושב במערך – אולי הוא הגדול ביותר.
if ac > max
    common = A[i-1];
return common;
```

**<u>נכונות</u>**:

בשלב א נמיין את המספרים, ולכן אם מספר מופיע יותר מפעם אחת – כל המופעים שלו יופיעו ברצף.

בשלב ב ננצל זאת כדי לספור כמה מופעים רצופים יש מכל איבר, ולאיזה איבר מספר המופעים הוא המקסימלי.

**<u>עלות:</u>**

*שלב א-* $\Theta(n\log n)$ (מיון).

(הערה – לא נשתמש במיון מניה, מכיוון שאנו לא יודעים את טווח הערכים של המספרים.)

*שלב ב-* $\Theta(n)$ (מעבר סידרתי על המערך.)

סה"כ - $\Theta(n\log n)$

## <u>חיפוש</u>

3) נתונים שני מערכים ממוינים A ו- B בעלי n איברים כל אחד. תאר אלגוריתם אשר מוצא את **חציון** שני המערכים בזמן O(log n).

<u>פתרון</u>

הרעיון – נשתמש בחיפוש בינארי. בכל שלב נשווה בין האיברים האמצעיים בכל מערך. ההשוואה תאפשר לנו לפסול מספר איברים שקטנים מהחציון <u>ואותו מספר</u> של איברים שגדולים מהחציון.

(לפי הגדרה, חציון הוא איבר בקבוצה שמחצית מן האיברים בה גדולים או שווים לו ומחצית מן האיברים קטנים או שווים לו. לכן, אם השמטנו מספר שווה של איברים קטנים וגדולים, אזי החציון של המספרים שנותרו הוא החציון של כל המספרים שהיו בהתחלה.)

נגדיר את החציון של <u>מערך זוגי</u> באורך 2k להיות האיבר במקום ה-k (כלומר, הקטן מבין שני "האמצעיים")**.**

**<u>קוד</u>:**

```
int median(A1, A2, l1, r1, l2, r2)
                    /* Given arrays A1,A2, and the left and right edges of each.  */
{
  1. if  (l1 == r1)
  2.      return min{A1[l1], A2[l2]}
  3. m1 = (r1 + l1)/2
  4. m2 = (r2 + l2)/2
  5. if  (A1[m1] == A2[m2])
  6.      return (A1[m1])
  7. if  (A1[m1] > A2[m2])
  8.      { r1 = m1
  9.        l2 = m2 + (r2 − l2)%2
  10.     }
  11. if  (A1[m1] < A2[m2])
  12.     { r1 = m1 + (r1 − l1)%2
  13.       l2 = m2
  14.     }
  15.     return median(A1, A2, l1, r1, l2, r2)
}
```

**<u>הסברים</u>**

שורות 1-2: תנאי עצירה – אם המערכים באורך 1 החזר את האיבר הקטן יותר.

שורות 3-4: מציאת האמצע של כל מערך (החציון של כל אחד מהמערכים).

שורות 5-6: אם החציונים של שני המערכים שווים – אזי הם החציון של כל האיברים (מתקיים שמחצית מן האיברים גדולים או שווים לו, ומחצית מהם קטנים או שווים לו), לכן נחזיר אותו.

שורות 7-10: אם החציון של מערך A1 **גדול** מהחציון של מערך A2, אזי כל האיברים שגדולים מהחציון של מערך A1 בוודאות <u>גדולים</u> מהחציון של כל האיברים, וכל האיברים שקטנים מהחציון של מערך A2 בוודאות <u>קטנים</u> מהחציון של כל האיברים. לכן נוכל להשמיט איברים אלה מהבעיה, ולפתור את הבעיה עם מערכים קטנים יותר.

שורות 11-14 : המקרה הסימטרי, כשהחציון של מערך A1 **קטן** מהחציון של מערך A2.

שורה 15: קריאה רקורסיבית לפתרון עבור הבעיה הקטנה יותר.

הערה – בשורה 9 ( וכן בשורה 12) אנו מוסיפים לחציון : 1 אם המערך באורך זוגי

0 אם המערך באורך אי זוגי.

המטרה היא לוודא שאנו מורידים משני המערכים את אותו מספר של איברים. (הוכחה בהמשך.)

**<u>נכונות</u>**: באינדוקציה על אורך המערכים n:

בסיס – n=1 : יש בסה"כ 2 איברים ולכן הקטן יותר הוא החציון.

צעד – נניח שעבור מערכים בגודל n<=2k האלג' מחזיר את החציון כנדרש.

נוכיח שגם עבור n=2k+1 ו- n=2k+2 האלג' מחזיר את החציון כנדרש:

נשים לב כי הביטוי (r2-l2)%2 יהיה שווה: 1 אם המערך באורך זוגי

0 אם המערך באורך אי זוגי.

<u>n=2k+1 :</u> החציונים של המערכים נמצאים באינדקס k+1. נניח בה"כ שהתנאי בשורה 7 מתקיים. אזי

במערך A1 נבצע: r1 = m1 = k+1 כלומר, נוריד בדיוק k איברים,

וכן במערך A2 נבצע: l2 = m2+(r2-l2)%2

l2 = m2+0 = k+1 כלומר, גם כאן נוריד בדיוק k איברים.

כעת נפעיל את הפונקציה על מערכים באורך k+1, ולכן לפי הנחת האינדוקציה יוחזר החציון הנכון.

<u>n=2k+2 :</u> החציונים של המערכים נמצאים באינדקס k+1. נניח בה"כ שהתנאי בשורה 7 מתקיים. אזי

במערך A1 נבצע: r1 = m1 = k+1 כלומר, נוריד בדיוק k+1 איברים,

וכן במערך A2 נבצע: l2 = m2+(r2-l2)%2

l2 = m2+1 = k+2 כלומר, גם כאן נוריד בדיוק k+1 איברים.

כעת נפעיל את הפונקציה על מערכים באורך k+1, ולכן לפי הנחת האינדוקציה יוחזר החציון הנכון.

**<u>עלות</u>**: בכל שלב אורך המערכים קטן פי 2 (כמו בחיפוש בינארי), ולכן זמן הריצה במקרה הגרוע יהיה:

$$T(n) = T(\frac{n}{2}) + \Theta(1) = \Theta(\log n)$$

(לפי מאסטר )

אבל יתכן שנמצא את החציון מהר יותר – אם התנאי בשורה 5 יתקיים, ולכן נוכל לומר שזמן הריצה

$O(\log n)$