**ממ"ן 13**

**שאלה 1**

נתון סדרת מספרים:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 30 | 29 | 25 | 21 | 19 | 15 | 11 | 9 | 5 | 1 | 100 | 90 | 80 | 70 | 60 |

נראה את השינויים המתבצעים בסדרת המספרים במיון QuickSort .

הערך j - בכל שלב הערך יהיה שווה למספר התאים הכחולים + 1.

הערך i - בכל שלב הערך יהיה שווה למספר התאים הירוקים.

הערך x - שווה לאיבר A[r] (אשר בהתחלה זהו ערך האיבר האחרון בסדרה ).

בכל שלב מתבצע השוואה בין הערך בנקודה j, לבין x.

א) רק אם A[j]<=x אז:

1. מתווסף ל i אחד.

2. מבצעים החלפה בין A[i] לבין A[j]

ב) ומתווסף ל j אחד.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 30 | 29 | 25 | 21 | 19 | 15 | 11 | 9 | 5 | 1 | 100 | 90 | 80 | 70 | 60 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 30 | 29 | 25 | 21 | 19 | 15 | 11 | 9 | 5 | 1 | 100 | 90 | 80 | 70 | 60 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 30 | 29 | 25 | 21 | 19 | 15 | 11 | 9 | 5 | 1 | 100 | 90 | 80 | 70 | 60 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 30 | 29 | 25 | 21 | 19 | 15 | 11 | 9 | 5 | 1 | 100 | 90 | 80 | 70 | 60 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 30 | 29 | 25 | 21 | 19 | 15 | 11 | 9 | 5 | 1 | 100 | 90 | 80 | 70 | 60 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 30 | 29 | 25 | 21 | 19 | 15 | 11 | 9 | 5 | 1 | 100 | 90 | 80 | 70 | 60 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 30 | 29 | 25 | 21 | 19 | 15 | 11 | 9 | 5 | 60 | 100 | 90 | 80 | 70 | 1 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 30 | 29 | 25 | 21 | 19 | 15 | 11 | 9 | 70 | 60 | 100 | 90 | 80 | 5 | 1 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 30 | 29 | 25 | 21 | 19 | 15 | 11 | 80 | 70 | 60 | 100 | 90 | 9 | 5 | 1 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 30 | 29 | 25 | 21 | 19 | 15 | 90 | 80 | 70 | 60 | 100 | 11 | 9 | 5 | 1 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 30 | 29 | 25 | 21 | 19 | 100 | 90 | 80 | 70 | 60 | 15 | 11 | 9 | 5 | 1 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 30 | 29 | 25 | 21 | 60 | 100 | 90 | 80 | 70 | 19 | 15 | 11 | 9 | 5 | 1 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 30 | 29 | 25 | 70 | 60 | 100 | 90 | 80 | 21 | 19 | 15 | 11 | 9 | 5 | 1 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 30 | 29 | 80 | 70 | 60 | 100 | 90 | 25 | 21 | 19 | 15 | 11 | 9 | 5 | 1 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 30 | 90 | 80 | 70 | 60 | 100 | 29 | 25 | 21 | 19 | 15 | 11 | 9 | 5 | 1 |

האלגוריתם סיים לעבור על כל הסדרה,מבצעים החלפה אחרונה בין איבר ה Pivot (A[r]) ל A[i+1] , והאיבר ה Pivot נמצא עכשיו במקומו - כלומר כל האיברים אשר לפניו בסדרה קטנים ממנו, וכל האיברים אשר אחריו בסדרה גדולים ממנו:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 100 | 90 | 80 | 70 | 60 | 30 | 29 | 25 | 21 | 19 | 15 | 11 | 9 | 5 | 1 |

בשלב זה, נחלק את הסדרה ל 2 סדרות, סדרת המספרים הכחולים, וסדרת המספרים הירוקים.

ושוב נפעיל את האלגוריתם על כל אחת מהסדרות בנפרד, כך נמשיך בצורה רקורסיבית עד שהכל יהיה ממויין וכל איבר יהיה במקומו.

במקרה שלנו, שני הסדרות ישארו ללא שינוי מכיוון שכבר ממויינות - האלגוריתם אמנם ימשיך לחלק את הסדרות (בכך שיחלק אותם כל פעם לאותה סדרה ללא האיבר האחרון ) אך הסדרות תשארנה ללא שינוי - והינה מיינו את הסדרה.מה יהיה סדר הקריאות? מה יהיו p ו-r. 3-

**שאלה 2**

**סעיף א'**

נשנה את השורה הראשונה באלגוריתם quicksort כך שהקריאה הרקורסיבית תתבצע אם ורק אם ההפרש בין קצוות המערך קטן מ k. כלומר:

OtherQUICKSORT(A,p,r,k)

1. if (p-r<k) then

2. q<-- Partition(A,p,r,k)

3. OtherQUICKSORT (A,p,q-1,k)

4. OtherQUICKSORT (A,q+1,r,k)

נבדוק מהו זמן הריצה של אלגוריתם זה במקרה הטוב ביותר:

במקרה הטוב ביותר, כל קריאה לשגרה Partition מחלקת את המערך הנתון באותה קריאה ל 2 תתי מערכים בגודל n/2 לכל היותר. ולכן מתקבל הנוסחת נסיגה הבאה:

*כלומר:*

ובשורה ה i :

לכל מתקיים , שכן במצב זה האלגוריתם מבצע מספר קבוע של פעולות, ולכן:

נדרוש:

, כלומר --> --> ולכן

ו

ונקבל :

כלומר, זמן ריצה של האלגוריתם במקרה הטוב ביותר הינו:

נבדוק מהו זמן הריצה של אלגוריתם זה במקרה הגרוע ביותר:

במקרה הגרוע ביותר שגרת האלגוריתם מחלקת את התת מערך לאזור המכיל n-1 איברים, ולאזור המכיל 0 איברים. מכיוון שהקריאה הרקורסיבית על מערך בגודל 0 חוזרת ללא ערך נוכל להגיד ש :

ונקבל את הנוסחת נסיגה הבאה:

כלומר:

ובשורה ה i:

לכל מתקיים , שכן במצב זה האלגוריתם מבצע מספר קבוע של פעולות, ולכן:

נדרוש:

*כלומר :*  , כך ש ונקבל :

כלומר, זמן ריצה של האלגוריתם במקרה הגרוע ביותר הינו:

ולכן, האלגוריתם החדש OtherQUICKSORT יעיל יותר מהאלגוריתם QUICKSORT המקורי גם עבור הזמן ריצה הטוב ביותר וגם עבור הזמן ריצה הגרוע ביותר.

**סעיף ב'**

המקרה הטוב ביותר של אלגוריתם 'מיון הכנסה' מתרחש כאשר הקלט המתקבל הוא ממוין לחלוטין, ואז כידוע זמן הריצה שלו הוא . לכן במידה והמערך הכמעט ממוין עם שגיאה בגודל k יהיה ממוין לחלוטין אחרי ה OtherQUICKSORT, אז יעילות מיון ההכנסה תהיה .

המקרה הגרוע ביותר של 'מיון הכנסה' מתרחש כאשר הקלט אינו ממוין או ממוין בסדר הפוך, ואז כידוע זמן הריצה שלו הוא .

ישנם לכל היותר קטעים לא ממוינים , אם בכל קטע כזה מיון ההכנסה יעבוד בזמן הגרוע ביותר, אז בכל קטע יתבצע מיון בגודל ( כי גודל כל קטע לא ממויין כזה הוא לכל היותר k ),

אז יעילות המיון תיהיה:

**שאלה 3**

**סעיף א'**

החלוקה בשלב הראשון מתבצעת בזמן , והיא משנה את המערך ל:

מכאן שהשגרה מבצעת קריאה רקורסיבית על 2 תתי מערכים: [1] ו [5,6,...,n,3,4].

התת מערך [1] כמו כבר ממויין, נתבונן בתת מערך [5,6,...,n,3,4], נשים לב שבכל קריאה רקורסיבית התת מערך יתפרק לתת מערך בעל איבר אחד, ותת מערך בעל שאר האיברים, כלומר נוסחת הנסיגה היא:

ובשורה ה i:

נדרוש , כלומר : , כך שמתקיים : , ונקבל:

וכמובן שנקבל:

**סעיף ב'**

החלוקה בשלב הראשון מתבצעת בזמן , והיא משנה את המערך ל:

מכאן שהשגרה מבצעת קריאה רקורסיבית על 2 תתי מערכים: [2,1] ו .

מיון המערך בעל 2 האיברים זניח, נתבונן במערך . נשים לב שמעתה ואילך בכל קריאה רקורסיבית התת מערך יתפרק לתת מערך בעל איבר אחד, ותת מערך בעל שאר האיברים, כלומר נוסחת הנסיגה משלב זה היא שוב:

וכמו שהראינו בסעיף קודם, נקבל:

**שאלה 4**

**סעיף א'**

הגדרת החציון - איברים בסדרה קטנים ממנו, ו איברים בסדרה גדולים ממנו.

כלומר, בהינתן החציון, ניתן להשוות כל אחד מהאיברים בסדרה לחציון ב פעולות השוואה, כך ש:

* אם האיבר קטן מהחציון - הכנס לסדרה s1.
* אם האיבר גדול מהחציון - הכנס לסדרה s2.

נתון כי ניתן למצוא את החציון בעזרת השוואות, כלומר סדרה בעלת איברים ניתנת לחצייה בעזרת השוואת.

**סעיף ב'**

לאחר שמבצעים חציה על סדרה בעלת איברים, החציון הוא האיבר הגדול ביותר הנמצא ב s1.

ב s1 ישנם איברים, ביניהם ניתן להשוות ולמצוא את המקסימום ע"י השוואות.

נתון כי סדרת בעלת איברים ניתנת לחצייה בעזרת השוואות, לכן ניתן למצוא את החציון של סדרה בעלת איברים בעזרת השוואות.

**שאלה 5**

נתונה סדרה של מספרים ממשיים ( ):

נסמן:

**סעיף א'**

נניח בשלילה שלכל מתקיים:

נסמן ב bi את האיבר ה i בגודלו בסדרה (, i טבעי ).

לכן מתקיימים גם כל הביטויים הבאים:

......

נחבר את כל הביטויים ונקבל:

*כלומר:*

*אבל הוא האיבר הקטן ביותר בסדרה, ו הוא האיבר הגדול ביותר בסדרה, כלומר מתקיים:*

*ו כלומר ע"פ המשוואה יוצא:*

בסתירה!

לכן קיימים המקיימים:

**סעיף ב'**

Lowest\_Difference(A,p,r)

1. if (p=r-1) then

2. return (A[p],A[r])

3. m <-- MINIMUM(A)

4. M <-- MAXIMUM(A)

5. q <-- CROSSING (A,p,r,) (\*)

6. if (M-A[q]<A[q]-m) then

7. return Lowest\_Difference(A,q,r)

8. else

9. return Lowest\_Difference(A,p,q)

CROSSING(A,p,r,

פעולת חציה כפי שמוגדרת בשאלה 4.

הוכחנו בשאלה 4 שפעולת החצייה שקולה למציאת החציון.

ידוע שמציאת חציון מתבצע בזמן לכן גם פעולת החציה תתבצע בזמן .

השגרה תחצה את המערך ל 2 תתי מערכים (בעזרת Partioton) כך שכל האיברים A[p,q-1] קטנים או שווים ל A[q], וכל האיברים A[q+1,r] יהיו גדולים מ A[q].

לבסוף - השגרה תחזיר את החציון המבוקש.

**נכונות השגרה:**

הביטוי בעצם מייצג את ההפרש הממוצע בין איברים בסדרה ממויינת.

בכל קריאה רקורסיבית, אנו מוצאים את המינימום בתת מערך, ואת המקסימום בתת מערך (שורות 3-4), ומשתמשים ב CROSSING *על מנת למצוא את החציון ולסדר את המערך כך שכל האיברים הקטנים או שווים לחציון נמצאים משמאלו, וכל האיברים הגדולים ממנו נמצאים מימינו (שורה 5).*

*לאחר מכן, אנו בודקים עבור 2 תתי המערכים, איזה הפרש בין המקסימום למינימום הוא קטן יותר (שורה 6 )- הפרש זה מייצג את* ההפרש הממוצע בין איברים (בסדרה ממויינת) בתת מערך זה - כלומר, בקבוצה שבה ההפרש קטן יותר, קיימים 2 איברים בקבוצה זאת אשר הפרשם קטן מ .

ולכן נבצע את הקריאה הרקורסיבית רק על תת מערך זה, ובכך נקטין את הבעיה פי 2 כל פעם.

תנאי העצירה מתקיים כאשר הגענו למערך המכיל רק 2 איברים, במקרה כזה, הגענו להפרש איברים אשר קטנים מהממוצע, ולכן יוחזרו 2 האיברים בתת מערך זה אשר מקיימים:

**יעילות האלגוריתם:**

*שורות 3-5 מתבצעות ביעילות של כל אחת, בנוסף, השגרה דואגת לכך שבכל קריאה רקורסיבית אנחנו מטפלים רק בחצי המערך הנותר, ולכן נוסחת הנסיגה תהיה:*

נשתמש במשפט שיטת האב:

a=1 , b=2 ,

לכן מתקיים מקרה (3) שבו קיים כך ש: , ובנוסף:

קיים c<1 המקיים:

**( מתקיים לדוגמא עבור c=0.6 )**

ולכן נקבל:

מ.ש.ל

תמיר אביב,

305652000