שאלה 2

אנו מפעילים את השגרה $PARTITION(A, 1, n)$, בהנחה שאיברי המערך $A[1..n]$ שונים זה מזה.

(א) אם A ממוין מלכתחילה בסדר עולה, לולאת while מתבצעת פעם אחת. בתוך הלולאה, ההשוואה $A[j] \le x$ מתבצעת n פעמים, ההשוואה $A[i] \ge x$ פעם אחת, וההשוואה $i<j$ פעם אחת; סה"כ $n+2$ השוואות.

(ב) אם A ממוין מלכתחילה בסדר יורד, לולאת while מתבצעת פעמיים. במעבר הראשון, ההשוואה $A[j] \le x$ מתבצעת פעם אחת, ההשוואה $A[i] \ge x$ פעם אחת, וההשוואה $i<j$ פעם אחת; במעבר השני, ההשוואה $A[j] \le x$ מתבצעת פעם אחת, ההשוואה $A[i] \ge x$ מתבצעת $n-1$ פעמים, וההשוואה $i<j$ פעם אחת; סה"כ $n+4$ השוואות.

שאלה 3

השערה החדשה:

```
QUICKSORT-UNFINISHED (A, k, p, r)
   if p+k < r
      then q ← PARTITION (A, p, r)
           QUICKSORT-UNFINISHED (A, k, p, q)
           QUICKSORT-UNFINISHED (A, k, q+1, r)
```

השגרה מתבצעת באמצעות הקריאה:

QUICKSORT-UNFINISHED (A, k, 1, n)

המקרה הטוב ביותר: בכל קריאה $q = \frac{1}{2}(p+r)$;
מספר הקריאות הוא $\lceil \lg \frac{n}{k} \rceil$;
זמן הביצוע: $\Theta(n \cdot \lg \frac{n}{k})$

המקרה הגרוע ביותר: בכל קריאה $q = r-1$ או $q = 1$;
מספר הקריאות הוא $n-k-1$;
זמן הביצוע: $\Theta(n \cdot (n-k))$

(זה יכול להיות $\Theta(n)$ אם $n-k = \Theta(1)$
או $\Theta(n^2)$ אם $n-k = \Theta(n)$).

שאלה 4

(א) ממיינים את $\lceil\sqrt{n}\rceil$ האיברים האחרונים של A (בעזרת מיון-מיזוג או מיון-מהיר);
זמן הריצה הוא $O(\lceil\sqrt{n}\rceil^2) = O(n)$.

בשלב שני, ממזגים את שני התת-מערכים הממוינים $A[1..n-\lceil\sqrt{n}\rceil]$
ו- $A[n-\lceil\sqrt{n}\rceil+1..n]$. סכום אורכי התת-מערכים הוא n; לכן, המיזוג
מתבצע בזמן $O(n)$.

זמן הריצה של האלגוריתם הוא $O(n)$.

(ב) אם נמיין את $f(n)$ האיברים האחרונים בעזרת מיון-מיזוג ואחר כך נבצע
מיזוג של שני התת-מערכים, זמן הריצה יהיה $O(n+f(n)\cdot \lg f(n))$. כדי
להגיע לזמן ריצה לינארי, נבחר את הפונקציה $f(n)=n/\lg n$; במקרה זה,

$$O(n+f(n)\cdot \lg f(n)) =$$

$$O\left(n+\frac{n}{\lg n}\cdot(\lg n-\lg\lg n)\right) = O(n)$$

כל פונקציה אחרת $g(n)$, המקיימת את התנאי

$$O(n+g(n)\cdot \lg g(n)) = O(n)$$

חייבת להיות קטנה יותר אסימפטוטית מהפונקציה $n/\lg n$ (או שווה לה
אסימפטוטית); לדוגמא, $\sqrt{n}=o(n/\lg n)$.