



שאלה 2 (20 נקודות)

הערה: כל המספרים המופיעים בשאלה הם מספרים שלמים, ובכל המערכים האיברים שונים זה מזה. נאמר שמערך הוא סימטרי סביב k , אם לכל מספר x שבמערך גם הערכים $\{k + |x - k|, k - |x - k|\}$ הם ערכים במערך.

במילים אחרות, אם נצייר את כל האיברים במערך על גבי ציר, נקבל תמונה סימטרית סביב k . לדוגמה, המערך

1	11	5	13	7	9	3	15
---	----	---	----	---	---	---	----

הוא סימטרי סביב 8.

שימו לב: מערך סימטרי סביב k לא בהכרח ממוין.

נתון מערך A באורך n . ידוע כי קיים k שלם עבורו המערך A סימטרי סביב k . עליכם לכתוב פונקציה בשם `FindSymmetricK`, אשר מקבלת את המערך, ואת אורכו, ומחזירה את k , בסיבוכיות זמן $O(n)$ ובסיבוכיות מקום $O(1)$.

```
int FindSymmetricK(int *A, int n)
{
    int min, max, i;
    min = max = A[0];
    for (i = 1; i < n; i++) {
        min = (A[i] < min) ? A[i] : min;
        max = (A[i] > max) ? A[i] : max;
    }
    return (min+max) / 2;
}
```



ב. בהמשך לסעיף הקודם, נגדיר מערך **סופר-סימטרי** באופן רקורסיבי.

- כל מערך באורך 1 הוא סופר-סימטרי.
- מערך A באורך $n > 1$ הוא סופר-סימטרי, אם קיים k עבורו A סימטרי סביב k , וכן כל איברי A הקטנים מ k הם תת מערך סופר-סימטרי וכן כל איברי A הגדולים מ k הם תת מערך סופר סימטרי.

לדוגמא, המערך מסעיף א' הינו סופר-סימטרי, כי עבור $k=8$ נקבל תת מערך $\{1,5,3,7\}$ שהוא סופר סימטרי עבור $k=4$, ותת מערך נוסף $\{11,13,9,15\}$ שהוא גם סופר סימטרי סביב $k=12$.

הערה: המערכים בשאלה הם באורך שהוא חזקה של 2.
עליכם לכתוב פונקציה **רקורסיבית** בשם `SortSuperSymmetric` אשר מקבלת מערך סופר-סימטרי A באורך n , וממיינת אותו בסיבוכיות זמן $O(n)$ וסיבוכיות מקום $O(\log n)$.

לצורך המימוש ניתן להשתמש בפונקציה `pivot` (שחתימתה נתונה בהמשך), אשר מקבלת מערך A , את אורכו n , ומספר x , ומסדרת את המערך כך שכל האיברים שקטנים מ- x ימצאו לפני כל האיברים שגדולים מ- x . סיבוכיות הזמן של הפונקציה `pivot` היא $O(n)$, וסיבוכיות המקום $O(1)$.
כמו כן, ניתן להשתמש בפונקציה מסעיף א', גם אם לא מימשתם אותה.

```
void pivot(int *A, int n, int x);
```

```
void SortSuperSymmetric(int *A, int n)
{
    int i, k;
    if (n <= 1) {
        return;
    }
    k = FindSymmetricK(A, n);
    pivot(A, n, k);
    SortSuperSymmetric(A, n/2);
    for (i=0; i < n/2; i++)
        A[n-i-1] = A[i] + (k-A[i])*2;
}
```