

### לפותרים ב-Java

לפניך הפעולות sod ו-what המקבלות מערך a שאיבריו מטיפוס שלם, ממיון בסדר עולה, ומספר שלם k. לשתי הפעולות אותה טענת יציאה.

```
public static boolean sod(int[] a , int k)
{
    for (int i = 0; i < a.length-1; i++)
    {
        int j = i+1;
        while (j < a.length)
        {
            if (a[i] + a[j] == k)
                return true;
            j++;
        }
    }
    return false;
}
```

```
public static boolean what(int[] a , int k)
{
    int left = 0 , right = a.length-1;
    while (left < right)
    {
        if (a[left] + a[right] == k)
            return true;
        if (a[left] + a[right] < k)
            left++;
        else
            right--;
    }
    return false;
}
```

נתון מערך a: 

2	4	7	12	18
---	---	---	----	----

- א.** עקוב בעזרת טבלת מעקב אחר ביצוע הפעולה sod בעבור המערך הנתון a והמספר  $k = 11$ . רשום את הערך המוחזר.
- בטבלת המעקב יש לכלול עמודות בעבור:  $i, j, a[i], a[j]$ , ועמודה נוספת שבה יצוין אם התנאי שבפקודת if מתקיים או אינו מתקיים.
- ב.** עקוב בעזרת טבלת מעקב אחר ביצוע הפעולה sod בעבור המערך הנתון a והמספר  $k = 10$ . רשום את הערך המוחזר.
- בטבלת המעקב יש לכלול את העמודות שפורטו בסעיף א.
- ג.** מהי טענת היציאה של הפעולה sod ?
- ד.** מהי סיבוכיות זמן הריצה של הפעולה sod ? נמק את תשובתך.
- ה.** עקוב בעזרת טבלת מעקב אחר ביצוע הפעולה what בעבור המערך הנתון a והמספר  $k = 11$ . רשום את הערך המוחזר.
- בטבלת המעקב יש לכלול עמודות בעבור:  $left, right, a[left], a[right]$ , ושתי עמודות נוספות לכל אחת מפקודות if. בכל עמודה יצוין אם התנאי בפקודת if מתקיים או אינו מתקיים.
- ו.** מהי סיבוכיות זמן הריצה של הפעולה what ? נמק את תשובתך.
- ז.** מי מבין שתי הפעולות — sod או what — יעילה יותר? נמק את תשובתך.
- ח.** טענת הכניסה של הפעולות sod ו- what שונתה כך שאפשר להעביר אליה מערך a לא ממוין.

(1) האם טענת היציאה של הפעולה sod תשתנה? נמק את תשובתך.

(2) האם טענת היציאה של הפעולה what תשתנה? נמק את תשובתך.

לפניך הפעולות Sod ו־ What המקבלות מערך a שאיבריו מטיפוס שלם, ממוין בסדר עולה, ומספר שלם k. לשתי הפעולות אותה טענת יציאה.

```
public static bool Sod(int[] a , int k)
{
    for (int i = 0; i < a.Length-1; i++)
    {
        int j = i+1;
        while (j < a.Length)
        {
            if (a[i] + a[j] == k)
                return true;
            j++;
        }
    }
    return false;
}
```

```
public static bool What(int[] a , int k)
{
    int left = 0 , right = a.Length-1;
    while (left < right)
    {
        if (a[left] + a[right] == k)
            return true;
        if (a[left] + a[right] < k)
            left++;
        else
            right--;
    }
    return false;
}
```

נתון מערך a: 

2	4	7	12	18
---	---	---	----	----

א. עקוב בעזרת טבלת מעקב אחר ביצוע הפעולה Sod בעבור המערך הנתון a

והמספר  $k = 11$ . רשום את הערך המוחזר.

בטבלת המעקב יש לכלול עמודות בעבור:  $a[j]$ ,  $a[i]$ ,  $j$ ,  $i$ , ועמודה נוספת שבה יצוין אם

התנאי שבפקודות if מתקיים או אינו מתקיים.

ב. עקוב בעזרת טבלת מעקב אחר ביצוע הפעולה Sod בעבור המערך הנתון a

והמספר  $k = 10$ . רשום את הערך המוחזר.

בטבלת המעקב יש לכלול את העמודות שפורטו בסעיף א.

ג. מהי טענת היציאה של הפעולה Sod ?

ד. מהי סיבוכיות זמן הריצה של הפעולה Sod ? נמק את תשובתך.

ה. עקוב בעזרת טבלת מעקב אחר ביצוע הפעולה What בעבור המערך הנתון a

והמספר  $k = 11$ . רשום את הערך המוחזר.

בטבלת המעקב יש לכלול עמודות בעבור:  $a[right]$ ,  $a[left]$ ,  $right$ ,  $left$ ,

ושתי עמודות נוספות לכל אחת מפקודות if. בכל עמודה יצוין אם התנאי

בפקודות if מתקיים או אינו מתקיים.

ו. מהי סיבוכיות זמן הריצה של הפעולה What ? נמק את תשובתך.

ז. מי מבין שתי הפעולות — Sod או What — יעילה יותר? נמק את תשובתך.

ח. טענת הכניסה של הפעולות Sod ו־ What שונתה כך שאפשר להעביר אליהן

מערך a לא ממוין.

(1) האם טענת היציאה של הפעולה Sod תשתנה? נמק את תשובתך.

(2) האם טענת היציאה של הפעולה What תשתנה? נמק את תשובתך.