

נתון מספר שלם וגדול מ-0.

שחלוף של מספר הוא מספר אחר באותו אורך, המורכב מאותן ספרות שמהן מורכב המספר הנוכחי, אך בסדר אחר. מספר המופיעים של כל ספרה במספר הנוכחי שווה למספר המופיעים של אותה ספרה במספר האחר.

לדוגמה: המספר 21611.

שחלופים אפשריים שלו הם המספרים: 12116, 62111.

נתונה המחלקה **DigitStorage** המייצגת מספר שלם וגדול מ-0. למחלקה יש תכונה אחת: מערך count בגודל 10 מטיפוס שלם. כל תא במערך מכיל את מספר המופיעים של הספרה במספר, שערכה שווה לערך המציין (האינדקס) של התא. לדוגמה: בעבור המספר 21611 המערך count ייראה כך:

	0	1	2	3	4	5	6	7	8	9
count	0	3	1	0	0	0	1	0	0	0

לפניך חלק מהמחלקה **DigitStorage** כתוב ב-Java וב-C#:

```
public class DigitStorage
{
    private int[] count = new int[10];
}
```

א. כתוב ב-Java או ב-C#, במחלקה **DigitStorage**, פעולה בונה שתקבל מספר שלם וגדול מ-0 ותאתחל את המערך count בהתאם.

ב. כתוב ב-Java או ב-C#, במחלקה **DigitStorage**, פעולה פנימית שתקבל עצם other מטיפוס **DigitStorage**. הפעולה תחזיר true – אם העצם other זהה לעצם הנוכחי, אחרת – הפעולה תחזיר false.

ג. כתוב ב-Java או ב-C# פעולה חיצונית שתקבל שני מספרים שלמים, גדולים מ-0 ושונים זה מזה. הפעולה תחזיר true – אם המספרים הם **שחלופים**, אחרת – הפעולה תחזיר false. עליך להשתמש בפעולות שכתבת בסעיפים א-ב.