

תרגיל בית 3 – תכנות שפת C **אחראי תרגיל – ערן בורשטיין**

information about the inputs:

- In this exercise you will deal with arrays of integers.
- the input will be in the form of : "number1,number2,..."
- which means - every two numbers are separated with one comma (there is no comma in the end).
- a valid input for example: "3,4,7,-1,34"
- In all questions you can assume a maximal array length of 20 elements.
- good luck.

Note that the run of your code needs to be exactly the same as the examples!

Question 1

Given an array (arr[]), return the number of subarrays (of consecutive elements) having a minimum sum.

Input:

Array of integers.

Test cases:

2,-3,4,6,-2,-1

Min sum: -3

Number of subarrays: 2

```
2,-3,4,6,-2,-1
Min sum: -3
Number of subarrays: 2
```

Explanation: the Subarray with consecutive elements and minimum sum will be "-3".
The number of subarrays is 2.

Question 2

Given an array (arr[]) of positive integers. Return an array in which all the odd numbers are at the beginning (to the left) and all the even numbers are at the end (to the right). Return the minimum number of swaps required to bring the array to its final state.

Input:

Array of integers.

Test cases:

1,2,3,4,5,7

Number of swaps: 2

```
1,2,3,4,5,7  
Number of swaps: 2
```

7,3,2,8

Number of swaps: 0

```
7,3,2,8  
Number of swaps: 0
```

Explanation:

No swaps are needed because all odd numbers are on the left already.

Question 3

Given an array (arr[]) which contains N elements.

Find if there are numbers occurring more than once in the given array, if so return 1, else return -1.

Input:

Array of integers.

Test cases:

2,3,1,2,3

1

```
2,3,1,2,3
1
```

Explanation: 2 and 3 occur more than once in the given array.

0,3,1,2

-1

```
0,3,1,2
-1
```

Explanation: all the elements are different.

Question 4

Given an array `arr[]`, the user inputs non-negative integers representing the height of a tower made out of blocks. If the width of each block is 1, compute how much water can be trapped between the blocks during the rainy season.

Input:

Array of integers.

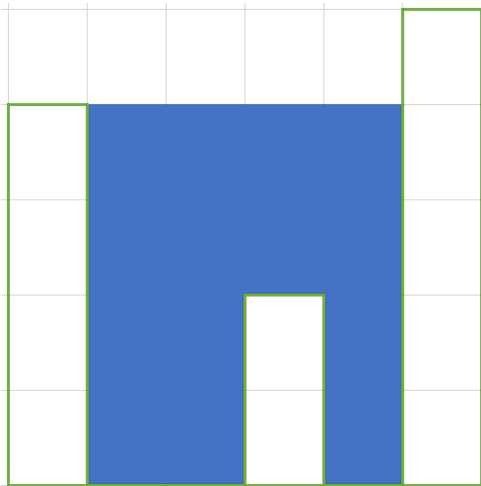
Examples:

1) Input: `arr[] = "4,0,0,2,0,5"`

```
4,0,0,2,0,5
14
```

Output: 14

Explanation:



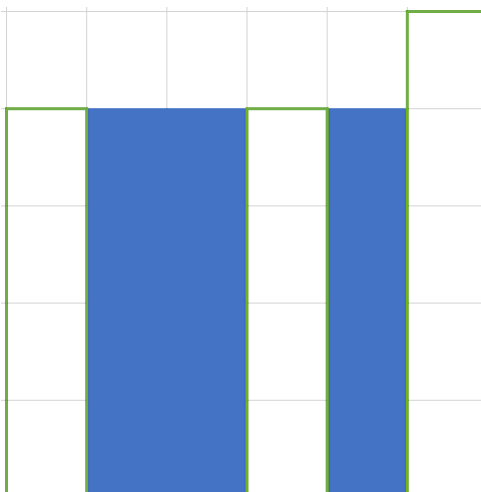
total trapped water = $4+4+2+4 = 14$

2) Input: `arr[] = "4,0,0,4,0,5"`

```
4,0,0,4,0,5
12
```

Output: 12

Explanation:

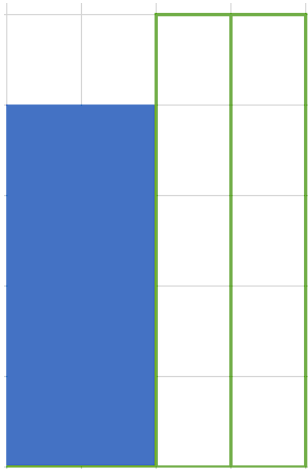


total trapped water = $4+4+0+4 = 12$

3) Input:arr[] = "0,0,5,5"

Output:0

Explanation:



No water will be trapped!

```
0,0,5,5
0
```

נהלים

קונבנציות

- יש לאתחל את כל המשתנים **בתחילת** כל בלוק קוד בתכנית באופן מסודר וקריא.
- Main תמיד יוגדר כזו `int` ויהיה לו ערך החזרה.
- מתן שמות משמעותיים למשתנים. לדוגמא, אם אגדיר משתנה מסוג `int` ששומר ערך של מניה, אקרא לו `stock` או `share` ולא `a` או `b`.
- כל פונקציה תכיל מקסימום 50 שורות קוד.
- יש לצרף תיעוד לקוד, אך בעיקר לשורות עם משמעות באלגוריתם. (כלומר אין צורך לתעד שורות טריוויאליות כמו `i=i++`).
- יש לשמור על מבנה מסודר של קוד, עם רווחים מתאימים לכל בלוק קוד.

פורמט הגשה

אנא הקפידו על נהלי ההגשה במדויק.

לתרגילים שלא יעמדו בדרישות ההגשה המצוינות תחת סעיף זה יורד ציון של 10 נקודות החל מתרגיל 2 (בתרגיל 1 יופחתו רק 5 נקודות על כל טעות כדי לבוא לקראתכם).

- ההגשה תתבצע באתר הקורס ב Moodle - תחת הקישור "תיבת ההגשה" שעל יד קובץ ההנחיות של אותו תרגיל הבית.
- לכל שאלה, תפתחו פרויקט חדש, כאשר תגישו רק את קובץ ה C מתוך הפרויקט קובץ ה-C ייקרא כפי שניתן לכם בשלד לכל תרגיל – **אין לשנות את שם הקובץ או ההדפסות המופיעות בשלד!**
- לדוגמא בתרגיל 1 תקבלו 2 קבצים בשמות `hw1_q1.c` ו `hw1_q2.c`, ואתם תצטרכו למלא את הקבצים האלו ולהריץ אותם.
- ההגשה הסופית תהיה קובץ Zip שמכיל את כלל קבצי הc שכתבתם. שם קובץ הzip יהיה בפורמט הבא:

`HW<מספר תעודת הזהות שלכם>_<מספר תרגיל הבית>.zip`

כלומר עבור תרגיל 1 תגישו קובץ zip ששמו `HW1_123456789.zip` (במקום הספרות 1-9 יופיע מספר תעודת הזהות שלכם) כאשר הzip יכיל 2 קבצים: `hw1_q1.c` ו `hw1_q2.c`.

בהצלחה,
צוות הקורס.