# Упражнения: Масиви

Тествайте решението в Judge: <https://judge.softuni.org/Contests/4146/01-Arrays-Advanced>.

## Завъртане на масиви

Напишете програма, която получава **масив** и **брой ротации**, които трябва да изпълните (първият елемент отива накрая). Отпечатайте получения масив.

### Примери

|  |  |
| --- | --- |
| **Вход** | **Изход** |
| 51 47 32 61 21  2 | 32 61 21 51 47 |
| 32 21 61 1  4 | 32 21 61 1 |
| 2 4 15 31  5 | 4 15 31 2 |

## Топ числа

Напишете програма, която **намира топ числата** в масива. Едно число е **топ**, ако е **по-голямо** от всички елементи **от дясната му страна**.

### Примери

|  |  |
| --- | --- |
| **Вход** | **Изход** |
| 1 4 3 2 | 4 3 2 |
| 14 24 3 19 15 17 | 24 19 17 |
| 27 19 42 2 13 45 48 | 48 |

## Еднакви суми

Напишете програма, която определя дали **съществува елемент в масива**, така че **сумата на елементите отляво** да е **равна** на **сумата на елементите отдясно (може да има не само един 1 такъв елемент)**. Ако **няма елементи от ляво/дясно**, тяхната **сума се зачита за 0**. Отпечатайте **индекса**, който отговаря на условието, или "**no**" ако няма такъв индекс.

### Примери

|  |  |  |
| --- | --- | --- |
| **Вход** | **Изход** | **Обяснения** |
| 1 2 3 3 | 2 | В a[2] -> лява сума = 3, дясна сума = 3  a[0] + a[1] = a[3] |
| 1 2 | no | В a[0] -> лява сума = 0, дясна сума = 2  В a[1] -> лява сума = 1, дясна сума = 0  Не съществува такъв индекс |
| 1 | 0 | В a[0] -> лява сума = 0, дясна сума = 0 |
| 1 2 3 | no | Не съществува такъв индекс |
| 10 5 5 99 3 4 2 5 1 1 4 | 3 | В a[3] -> лява сума = 20, дясна сума = 20  a[0] + a[1] + a[2] = a[4] + a[5] + a[6] + a[7] + a[8] + a[9] + a[10] |

## Съкращаване на масив от числа

Напишете програма, която прочита масив от цели числа и ги **съкращава** чрез сумиране на съседни двойки елементи, докато се получи едно **цяло число**. Например, ако имаме 3 елемента {2, 10, 3}, ние сумираме първите две числа и вторите две числа и получаваме {2+10, 10+3} = {12, 13}, след това отново сумираме всички съседни елементи и получаваме {12+13} = {25}.

### Примерен вход

|  |  |  |
| --- | --- | --- |
| **Вход** | **Изход** | **Обяснения** |
| 2 10 3 | 25 | 2 10 3 à 2+10 10+3 à 12 13 à 12 + 13 à 25 |
| 5 0 4 1 2 | 35 | 5 0 4 1 2 à 5+0 0+4 4+1 1+2 à 5 4 5 3 à 5+4 4+5 5+3 à 9 9 8 à 9+9 9+8 à 18 17 à 18+17 à 35 |
| 1 | 1 | Има само едно число |

### Насоки

Докато имаме повече от 1 елемент в масива nums[] повтаряме следните стъпки:

* Създаваме нов масив condensed[] с размер nums.Length-1.
* Събираме числата от nums[] в condensed[]
  + condensed[i] = nums[i] + nums[i+1]
* nums[] = condensed[]

Този процес може да се види на илюстрацията:

A picture containing diagram

Description automatically generated

A picture containing graphical user interface

Description automatically generated

## Криптиране, сортиране, и отпечатване на масив

Напишете програма, която чете редица от низове от конзолата. Криптирайте всеки низ като сумирате:

* Кода на всяка **гласна умножен по дължината на низа**
* Кода на всяка **съгласна разделен на дължината на низа**

**Сортирайте редиците** по азбучен ред. Отпечатайте тяхната криптация.

На първия ред ще получите броя на редиците от низове.

На следващите редове ще получите редиците

### Примери

|  |  |  |
| --- | --- | --- |
| **Вход** | **Изход** | **Обяснения** |
| 4  Peter  Maria  Katya  Todor | 1032  1071  1168  1532 | Peter = 1071  Maria = 1532  Katya = 1032  Todor = 1168 |
| 3  Sofia  London  Washington | 1396  1601  3202 | Sofia = 1601  London = 1396  Washington = 3202 |