# Допълнителни упражнения: Списъци

Тествайте решението в Judge: <https://judge.softuni.bg/Contests/3171/Additional-Exercises>.

## Еднакви числа

Напишете програма, която да събира всички съседни числа с еднаква сума в списъкът отляво надясно.

* След като **сумирате** **двете числа** , **резултатът** може да бъде равен на **съдеда** на второто число и трябва да се **сумира също** (както е показоно по-долу).
* Винаги сумирайте **най-левите два равни съседа** (ако има няколко равни двойки съседи)

Програмата свършва, когато няма повече съседни числа с еднакви стойности. Накрая отпечатайте крайният резултат.

### Примери

|  |  |  |
| --- | --- | --- |
| **Вход** | **Изход** | **Обяснения** |
| 3 3 6 1 | 12 1 | **3 3** 6 1 à **6 6** 1 à 12 1 |
| 8 2 2 4 8 16 | 16 8 16 | 8 **2 2** 4 8 16 à 8 **4 4** 8 16 à **8 8** 8 16 à 16 8 16 |
| 5 4 2 1 1 4 | 5 8 4 | 5 4 2 **1 1** 4 à 5 4 **2 2** 4 à 5 **4 4** 4 à 5 8 4 |

### Насоки

Първо четем списък от числа.

Text

Description automatically generated with medium confidence

Минаваме през всички елементи. Проверяваме дали числото със сегашният индекс и следващото са еднакви. Ако е така, нулираме for-цикъла и започваме отначало. В противен случей не правим нищо.

Text

Description automatically generated

Накрая отпечатваме всички числа **разделени с интервал**.



## Манипулация на списък

Напишете програма, която чете списък от числа. Докато не получите **"end",** ще получавате следните възможни **команди**:

**Add {число}:** Добавяне на число в края на списъка.  
**Remove {число}:** Премахване на число от списъка.  
**RemoveAt {индекс}:** Премахване на число с определен индекс.  
**Insert {число} {индекс}:** Вмъкване на число на определен индекс.

**Бележка: Не всичко ще бъде валидно!**

Когато получите **"end",** принтирате **финалния** списъка (**числата трябва да бъдат разделени с интервал**).

### Примери

|  |  |
| --- | --- |
| **Вход** | **Изход** |
| 4 19 2 53 6 43  Add 3  Remove 2  RemoveAt 1  Insert 8 3  end | 4 53 6 8 43 3 |

### Насоки

Първо прочитаме списъка от конзолата.

Text

Description automatically generated

* **Разделяме** низа получен на първия ред от конзолата. След това **минаваме през всеки елемент** и го парсваме към **число**.
* Това връща **IEnumarable<int>** (**колекция** от числа) и го запазваме под формата на списък.

След входа използваме while-цикъл и switch case за различните команди.

Graphical user interface, text, application

Description automatically generated

* Спираме цикъла, когато реда е “end”, в противен случай **разделяме** входа на низвое в **tokens**.

A picture containing table

Description automatically generated

Имплементираме всички **команди**.

Text

Description automatically generated

* За всички команди с **изключение** на **"Insert"** , **tokens[1]** е винаги **число/индекс**.
* За **"Insert"** командата ние получаваме **число и индекс** (**tokens[1], tokens[2]**).

Накрая отпечатваме числата, разделени с интервал.



## Манипулация на списъци (Advanced)

В тази задача ще трябва да имплентираме по-сложни команди на списъка(**разширяване на предходната задача**). Отново ще четете списък и команди, докато не получите **"end":**

**Contains {число}** – Проверявате дали число съществува в списъка - Ако е така принтирайте **"Yes**", в противен случай **"No such number ".**

**PrintEven** – принтирайте **всички четени числа разделени с интервал.**

**PrintOdd** – принтирайте **всички нечетени числа разделени с интервал.**  
**GetSum** – принтирайте **сумата на всички числа**.  
**Filter {condition} {number}** – отпечатайте всички числа, които отговарят на **даденото условие**. Условията могат да бъдат **'<', '>', ">=", "<="**.

След командата **"end"** принтирайте списъка, ако има някакви промени от оригиналния списък. **Промените** се правят само от командите от **предходната задача**.

### Примери

|  |  |
| --- | --- |
| **Вход** | **Изход** |
| 2 13 43 876 342 23 543  Contains 100  Contains 543  PrintEven  PrintOdd  GetSum  Filter >= 43  Filter < 100  end | No such number  Yes  2 876 342  13 43 23 543  1842  43 876 342 543  2 13 43 23 |