# Упражнения: Списъци – манипулации

## Въвеждане на списък от конзолата

Въведете списък от цели числа и го изведете в конзолата.

1. Стартирайте Microsoft Visual Studio 2022.
2. Създайте нов проект, като изберете: Create New Project

Graphical user interface, text, application

Description automatically generated

1. Изберете шаблон на проекта: [C#] 🡪 [Windows] 🡪 [Console] 🡪 [Console App]

Graphical user interface, application, Teams

Description automatically generated

1. Задайте име на проекта: **ListInputOutput**

Graphical user interface, text, application, email

Description automatically generated

1. Включете отметката: Do not use top-level statements

Graphical user interface, application

Description automatically generated

1. В метода Main() на файла Program.cs напишете следния програмен код:

int n = int.Parse(Console.ReadLine());

List<int> list = new List<int>();

for (int i = 0; i < n; i++)

list.Add(int.Parse(Console.ReadLine()));

for (int index = 0; index < list.Count; index++)

Console.WriteLine("list[{0}] = {1}", index, list[index]);

1. Стартирайте програмата с натискане на [Ctrl+F5].
2. Въведете брой на елементите в списъка – цяло число.
3. Въведете на отделни редове елементите за списъка.
4. Очаквайте от програмата подобен резултат:

Text

Description automatically generated

## Премахни числото

Въведете списък от цели числа и премахнете всички срещания в списъка на последното число. Елементите на списъка ще получите от единствен ред, разделени с интервали.

### **Примери**

|  |  |
| --- | --- |
| Вход | Изход |
| 3 4 1 5 1 5 2 1 | 3 4 5 5 2 |
| 7 3 8 3 5 3 7 3 | 7 8 5 7 |
| 2 2 8 2 5 2 3 2 | 8 5 3 |

### **Подсказски**

* Извлечете стойността на последния елемент. Той се намира на индекс равен на броя на елементите минус 1. Броят на елементите може да разберете чрез Count
* Докато елементът съществува, премахвайте първото му срещане чрез Remove

## Изтриване на отрицателни елементи

Въведете списък от цели числа, премахнете всички отрицателни числа от него и го изведете на конзолата в обратен ред. В случай, че в списъка не са останали елементи, изведете “empty”.

### **Примери**

|  |  |
| --- | --- |
| Вход | Изход |
| 10 -5 7 9 -33 50 | 50 9 7 10 |
| 7 -2 -10 1 | 1 7 |
| -1 -2 -3 | Empty |

### **Подсказки**

* Създайте нов празен списък за получения като резултат списък
* Обходете въведения списък отзад напред. Проверете всеки елемент и добавете неотрицателните елементи към списъка за резултат
* Накрая, изведете списъка резултат на единствен ред, разделен с интервали.

## Сливане на списъци

Напишете програма, която слива няколко списъка от числа.

* Списъците се разделят от ‘**|**’.
* Стойностите се разделят от интервали (‘’, един или няколко)
* Подредете списъците отзад напред, а техните стойности отляво надясно.

### **Примери**

|  |  |
| --- | --- |
| Вход | Изход |
| 1 2 3 |4 5 6 | 7 8 | 7 8 4 5 6 1 2 3 |
| 7 | 4 5|1 0| 2 5 |3 | 3 2 5 1 0 4 5 7 |
| 1| 4 5 6 7 | 8 9 | 8 9 4 5 6 7 1 |

### **Подсказки**

* Създайте нов празен списък за резултатите.
* Отделете входа чрез ‘**|**’ така че да се получи списък от низове.
* Обходете получения списък отдясно наляво.
  + За всеки низ в списъка: отделете елементите му чрез знака за интервал
  + Всеки един елемент, който е непразен низ, трябва да бъде добавен към списъка с резултата
* Изведете списъка с резултата

## Бомбички

Напишете програма, която въвежда поредица от числа и специално число - бомбичка с определена сила. Вашата задача е да детонирате всяко срещане на специалното число бомба и според нейната сила нейните съседи отляво и отдясно. Детонациите се изпълняват отляво надясно и всички детонирани числа изчезват. Най-накрая изведете сумата от оставащите елементи в поредицата.

### **Примери**

|  |  |  |
| --- | --- | --- |
| Вход | Изход | Коментари |
| 1 2 2 4 2 2 2 9  4 2 | 12 | Бомбичката е 4 със сила 2. След детонацията остават [1, 2, 9] със сума 12. |
| 1 4 4 2 8 9 1  9 3 | 5 | Бомбичката е 9 със сила 3. След детонацията оставаме с поредицата [1, 4], която има сума 5. Понеже 9 има само 1 съсед отдясно, ние го премахваме |
| 1 7 7 1 2 3  7 1 | 6 | Детонациите се изпълняват отляво надясно. Не можем да детонираме второто срещане на 7, понеже то вече е унищожено от първата детонация. Остават [1, 2, 3]. Тяхната сума е 6. |
| 1 1 2 1 1 1 2 1 1 1  2 1 | 4 | Оцветените числа изчезват в две последователни детонации. Оставащата предица е [1, 1, 1, 1]. Нейната сума е 4. |