# Упражнения: Списъци – сортиране

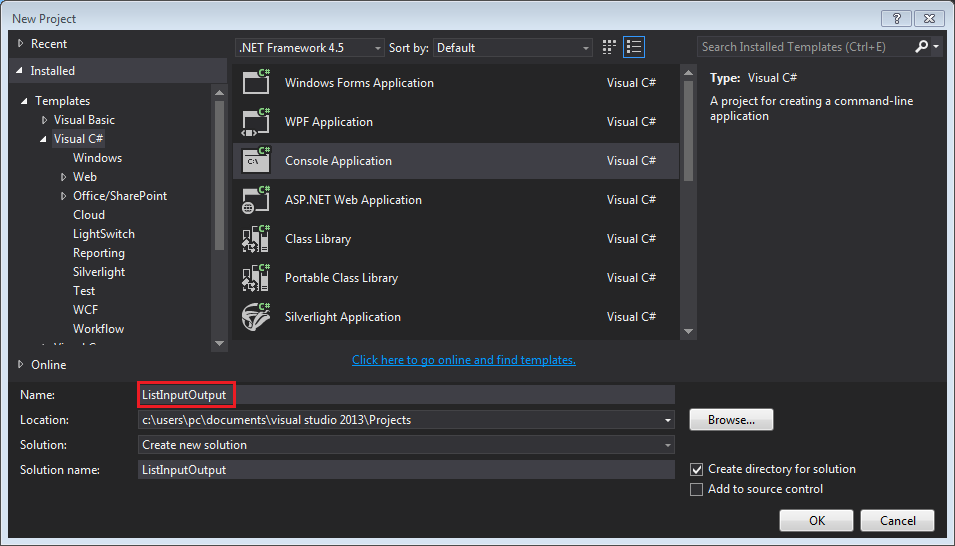
## Въвеждане на списък от конзолата

Въведете **списък от цели числа** и го изведете в конзолата

1. Стартирайте Visual Studio.
2. Създайте нов конзолен проект: [File]🡪 [New] 🡪 [Project].

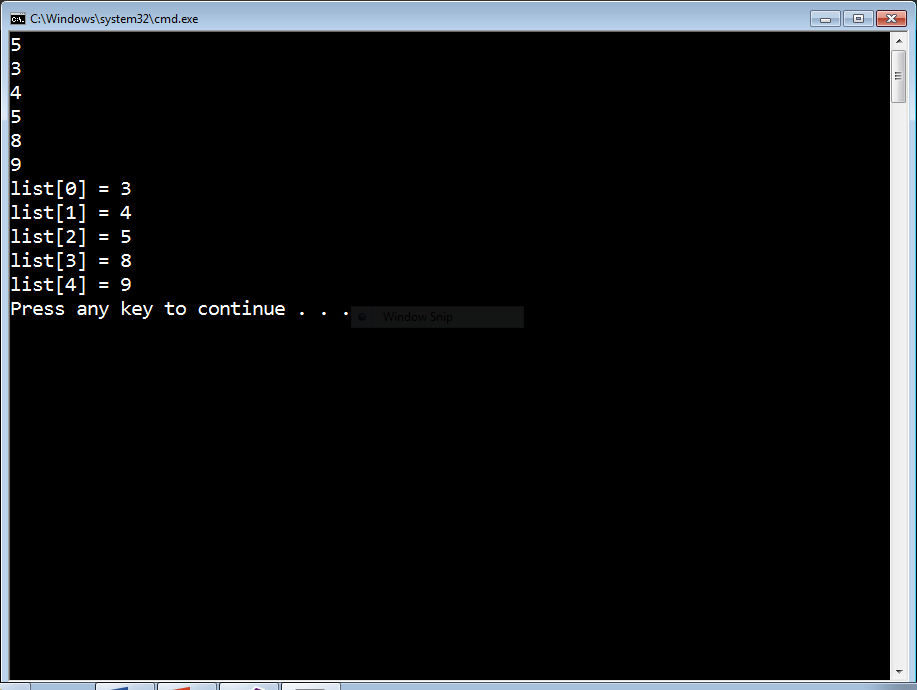


1. Изберете от диалоговия прозорец [Visual C#] 🡪 [Windows] 🡪 [Console Application] и дайте подходящо име на проекта, например “ListInputOutput”:



1. Намерете секцията Main(string[] args). В нея 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:  
   

## Сортиране на числа

Въведете **списък от цели числа** и го сортирайте.

### Примери

|  |  |
| --- | --- |
| **Вход** | **Изход** |
| 8 2 7 3 | 2 <= 3 <= 7 <= 8 |
| 1 1 | 1 <= 1 |
| 2 4 -9 | -9 <= 2 <= 4 |
| 1 -0.5 | -0.5 <= 1 |

## Числа квадрати

Въведете **списък от цели числа** и **изведете всички числа квадрати** от списъка в **нямалящ ред**. **Число квадрат** е цяло число, което е квадрат на друго цяло число. Например, 1, 4, 9, 16 са числа квадрати.

### Примери

|  |  |
| --- | --- |
| **Вход** | **Изход** |
| 3 **16 4** 5 6 8 **9** | 16 9 4 |
| 12 **1 9 4 16** 8 **25 49 16** | 49 25 16 16 9 4 1 |

### Подсказки

* За да разберете дали едно цяло число е “**число квадрат**”, проверете дали неговия корен квадратен е цяло число (такова че да няма дробна част):
  + if (√num == (int)√num) …
* За да подредите списъка от резултати в намалящ ред използвайте сортиране с ламбда функция:
  + squareNums.Sort((a, b) => b.CompareTo(a));

## Брой на числа

Въведете **списък от цели числа** в интервала [0…1000] и **ги изведете в нарастващ ред** заедно с **броя на срещанията им**.

### Примери

|  |  |
| --- | --- |
| **Вход** | **Изход** |
| 8 2 2 8 2 2 3 7 | 2 -> 4  3 -> 1  7 -> 1  8 -> 2 |
| 10 8 8 10 10 | 8 -> 2  10 -> 3 |

### Подсказки

1. Въведете елементите в масива от цели числа nums[]. Например: {8, 2, 2, 8, 2, 2, 3, 7}.
2. Сортирайте nums[] в нарастващ ред: {2, 2, 2, 2, 3, 7, 8, 8}. Сега намерете всички подредици от едни и същи числа.
3. **Обходете** числата отляво надясно. Пребройте колко пъти се среща всяко число
   * Започнете с count = 1.
   * Докато следващото число отдясно е **същото** като сегашното, **увеличавайте** count и продължете към следващото число.
   * Когато числото отдясно е **различно** (или няма друго число), **изведете** текущия елемент и неговия брой.
   * Продължете да обхождате от следващото число отдясно.