# Лабораторна работа: Набори и речници за напреднали

Задачи за [курса "C# Advanced" @ SoftUni](https://softuni.bg/trainings/4223/csharp-advanced-september-2023)   
Можете да проверите решенията си в [Judge](https://judge.softuni.org/Contests/1465/Sets-and-Dictionaries-Advanced-Lab)

## Речници

### Преброяване на същите стойности в масив

Създайте програма, която отчита в даден масив от двойни стойности броя на срещанията на всяка стойност.

#### Примери

|  |  |
| --- | --- |
| **Вход** | **Изход** |
| -2,5 4 3 -2,5 -5,5 4 3 3 -2,5 3 | -2,5 - 3 пъти  4-2 пъти  3-4 пъти  -5,5 - 1 пъти |
| 2 4 4 5 5 2 3 3 4 4 3 3 4 3 5 3 2 5 4 3 | 2-3 пъти  4-6 пъти  5-4 пъти   1. - 7 пъти |

### Средни оценки на студентите

Създайте програма, която чете **име** на ученик и неговите/нейните **оценки** и ги **добавя** към **записа на ученика** , след което **отпечатва** **имената** на учениците **с** оценките **и** средния им **успех** .

#### Примери

|  |  |
| --- | --- |
| **Вход** | **Изход** |
| 7  Йоан 5.20  Мария 5.50  Йоан 3.20  Мария 2.50  Сам 2.00  Мария 3.46  Сам 3.00 | Йоан -> 5,20 3,20 (ср.: 4,20)  Мария -> 5,50 2,50 3,46 (ср.: 3,82)  Сам -> 2,00 3,00 (ср.: 2,50) |
| 4  Влад 4.50  Петър 3.00  Влад 5.00  Петър 3.66 | Влад -> 4,50 5,00 (ср.: 4,75)  Петър -> 3,00 3,66 (ср.: 3,33) |
| 5  Георги 6.00  Георги 5.50  Георги 6.00  Йоан 4.40  Петър 3.30 | Джордж -> 6,00 5,50 6,00 (ср.: 5,83)  Йоан -> 4,40 (ср.: 4,40)  Петър -> 3,30 (ср.: 3,30) |

#### Съвети

* Използвайте **речник** ( низ 🡪 Списък<десетичен> )
* Проверете дали името **съществува,** преди да добавите оценка. Ако не, добавете го към речника.
* Преминете през всички **двойки ключ-стойност** в речника и отпечатайте резултатите. Можете да използвате .Average() метод за бързо изчисляване на средната стойност от списък.

### Най-големите 3 числа

Прочетете **списък с цели числа** и **отпечатайте най-големите 3 от тях** . Ако са по-малко от 3, отпечатайте всички.

#### Примери

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Вход** | **Изход** |  | **Вход** | **Изход** |
| 10 30 15 20 50 5 | 50 30 20 | 20 30 | 30 20 |

#### Съвети

* Прочетете масив от цели числа.
* **Подредете масива с помощта на LINQ заявка.**

**A picture containing text

Description automatically generated**

* **Отпечатайте първите 3 числа с for цикъл.**

### Продуктов магазин

Създайте програма, която разпечатва информация за **хранителните магазини** в София и **продуктите** , които **съхраняват** . Докато не бъде получена командата „ Ревизия “, ще получавате входни данни във формат: „{магазин}, {продукт}, {цена}“ . Имайте предвид, че ако получите **магазин** , който вече **имате** **получени** , трябва да **вземете** неговия **продукт** **информация** .

Вашият резултат трябва да бъде **подреден** по **име на магазин** и трябва да бъде във формат:

„{магазин}->

Продукт: {product}, Цена: {price}"

Забележка: Цената не трябва да бъде закръглена или форматирана.

#### Примери

|  |  |
| --- | --- |
| **Вход** | **Изход** |
| lidl, сок, 2.30  fantastico, ябълка, 1.20  кауфланд, банан, 1.10  фантастико, грозде, 2.20  Ревизия | fantastico->  Продукт: ябълка, Цена: 1.2  Продукт: грозде, Цена: 2.2  kaufland->  Продукт: банан, Цена: 1.1  lidl->  Продукт: сок, Цена: 2.3 |
| tmarket, фъстъци, 2.20  GoGrill, кюфтета, 3.30ч  GoGrill, HotDog, 1.40  tmarket, сладкиши, 2.20  Ревизия | GoGrill->  Продукт: кюфтета, Цена: 3.3  Продукт: HotDog, Цена: 1.4  tmarket->  Продукт: фъстъци, Цена: 2.2  Продукт: сладки, Цена: 2.2 |

### Градове по континенти и държави

Създайте програма, която чете **континенти** , **държави** и техните **градове,** поставете ги във **вложен речник** и ги **отпечатайте** .

#### Примери

|  |  |
| --- | --- |
| **Вход** | **Изход** |
| 9  Европа България София  Азия Китай Пекин  Азия Япония Токио  Европа Полша Варшава  Европа Германия Берлин  Европа Полша Познан  Европа България Пловдив  Африка Нигерия Абуджа  Азия Китай Шанхай | Европа:  България -> София, Пловдив  Полша -> Варшава, Познан  Германия -> Берлин  Азия:  Китай -> Пекин, Шанхай  Япония -> Токио  Африка:  Нигерия -> Абуджа |
| 3  Европа Германия Берлин  Европа България Варна  Африка Египет Кайро | Европа:  Германия -> Берлин  България -> Варна  Африка:  Египет -> Кайро |
| 8  Африка Сомалия Могадишу  Азия Индия Мумбай  Азия Индия Делхи  Европа Франция Париж  Азия Индия Нагпур  Европа Германия Хамбург  Европа Полша Гданск  Европа Германия Данциг | Африка:  Сомалия -> Могадишу  Азия:  Индия -> Мумбай, Делхи, Нагпур  Европа:  Франция -> Париж  Германия -> Хамбург, Данциг  Полша -> Гданск |

#### Съвети

* Използвайте **вложено** **речник** ( низ 🡪( Списък с речници <низ>) 🡪).
* Проверете дали континентът съществува, преди да добавите страната. Ако не, добавете го към речника.
* Проверете дали страната съществува, преди да добавите града. Ако не, добавете го към речника.
* Преминете през всички **двойки ключ-стойност** в речника и двойките ключ-стойност на стойностите и отпечатайте резултатите.

## Комплекти

### Записвайте уникални имена

Създайте програма, която ще вземе списък с **имена** и ще отпечата **само уникалните имена** в списъка.

#### Примери

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Вход** | **Изход** |  | **Вход** | **Изход** |  | **Вход** | **Изход** |
| 8  Джон  Алекс  Джон  Сам  Алекс  Алис  Петър  Алекс | Джон  Алекс  Сам  Алис  Петър |  | 7  Лайл  Брус  Алис  Истън  Шон  Алис  Шон  Петър | Лайл  Брус  Алис  Истън  Шон |  | 6  Роки  Роки  Роки  Роки  Роки  Роки | Роки |

#### Съвети

Можете да съхранявате имената в HashSet<string> да извлича само уникалните.

### Паркинг

Създайте програма, която:

* Записва **номер на автомобил** за всяка кола, която влиза на **паркинга.**
* Премахва **номер на кола** , когато колата напусне **паркинга.**

Входът ще бъде низ във формат: **"посока, номер на кола"** . Ще получавате команди до Команда **" КРАЙ ".** едадено.

Разпечатайте номерата на колите, които все още са на паркинга:

#### Примери

|  |  |
| --- | --- |
| **Вход** | **Изход** |
| IN , CA2844AA  IN , CA1234TA  ИЗВЪН , CA2844AA  IN , CA9999TT  IN , CA2866HI  ИЗВЪН , CA1234TA  IN , CA2844AA  ИЗВЪН , CA2866HI  IN , CA9876HH  IN , CA2822UU  КРАЙ | CA9999TT  CA2844AA  CA9876HH  CA2822UU |
| IN , CA2844AA  IN , CA1234TA  ИЗВЪН , CA2844AA  ИЗВЪН , CA1234TA  КРАЙ | Паркингът е празен |

#### Съвети

* Номерата на колите са **уникални.**
* Преди печат, **първо** **проверете** дали комплектът има елементи.

#### Решение

Можете да си помогнете с кода по-долу:



### СофтУни парти

#### Има купон в СофтУни . Поканени са много гости и те са два вида: VIP и Regular. Когато дойде гост, проверете дали той/тя съществува в някой от двата списъка с резервации.

Всички резервационни номера ще бъдат с дължина 8 знака.

Всички VIP номера започват с цифра.

Първо ще получите номерата на резервациите на гостите. Можете също да получите 2 възможни команди:

* „ПАРТИ“ – След тази команда ще започнете да получавате резервационните номера на хората, дошли на партито.
* „КРАЙ“ – Партито свърши и трябва да спрете програмата и да отпечатате подходящия резултат.

Накрая отпечатайте броя на гостите, които не са дошли на партито, и след това отпечатайте номерата на резервациите им. ВИП гостите трябва да са първи.

#### Примери

|  |  |  |  |
| --- | --- | --- | --- |
| **Вход** | **Изход** | **Вход** | **Изход** |
| 7IK9Yo0h  9NoBUajQ  Ce8vwPmE  SVQXQCbc  tSzE5t0p  ПАРТИ  9NoBUajQ  Ce8vwPmE  SVQXQCbc  КРАЙ | 2  7IK9Yo0h  tSzE5t0p | m8rfQBvl  fc1oZCE0  UgffRkOn  7ugX7bm0  9CQBGUeJ  2FQZT3uC  dziNz78I  mdSGyQCJ  LjcVpmDL  fPXNHpm1  HTTbwRmM  B5yTkMQi  8N0FThqG  xys2FYzn  MDzcM9ZK  ПАРТИ  2FQZT3uC  dziNz78I  mdSGyQCJ  LjcVpmDL  fPXNHpm1  HTTbwRmM  B5yTkMQi  8N0FThqG  m8rfQBvl  fc1oZCE0  UgffRkOn  7ugX7bm0  9CQBGUeJ  КРАЙ | 2  xys2FYzn  MDzcM9ZK |