# Речници

Тествайте задачите от тази тема в judge: <https://judge.softuni.bg/Contests/2669/Речници>

## Нечетни срещания

### Напишете програма, която извлича от поредица от думи всички елементи, които се срещат нечетен брой пъти (без значение от големината на буквите)

### Думите са въведени на един ред разделени с интервал

### Изведете получените думи с малки букви, по реда им на поява

### Примери

|  |  |
| --- | --- |
| **Вход** | **Изход** |
| Java C# PHP PHP JAVA C java | java, c#, c |
| 3 5 5 hi pi HO Hi 5 ho 3 hi pi | 5, hi |
| a a A SQL xx a xx a A a XX c | a, sql, xx, c |

### Подсказки

* Използвайте **речник** (string 🡪 int), за да преброите всички срещания на всяка дума (като в предната задача).
* Обходете всички **двойки ключ-стойност** в речника и запазете в резултатите всички **ключове**, които имат **нечетна стойност**.
* Изведете списъка с резултатите

## Брой на реалните числа

Въведете **списък от реални числа** и ги **изведете в нарастващ ред** заедно с броя на **срещанията им.**

**Примери**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Вход** | **Изход** |  | **Вход** | **Изход** |  | **Вход** | **Изход** |
| 8 2.5 2.5 8 2.5 | 2.5 -> 3  8 -> 2 | 1.5 5 1.5 3 | 1.5 -> 2  3 -> 1  5 -> 1 | -2 0.33 0.33 2 | -2 -> 1  0.33 -> 2  2 -> 1 |

**Подсказки**

* Използвайте **SortedDictionary<double,** **int>** с име **counts**.
* Обходете всяко число **num** и увеличете **counts[num]** (когато **num** съществува в речника) или присвоете **counts[num]** = **1** (когато **num** не съществува в речника).
* Обходете всяко едно от числата **num** в речника (**counts.Keys**) и изведете числото **num** и броя на срещанията му **counts[num]**.

## Телефонен указател

Напишете програма, която получава информация от конзолата за **хора** и техните **телефонни номера**. Всеки запис трябва да има само едно име и телефон (и двете се пазят в низ).

На всеки ред ще получите някоя от следните команди:

* **A {име} {телефон}** – добавя записа към телефонния указател. В случай че се добавя име, което вече съществува в указателя трябва да смените съществуващия номер с новия.
* **S {име}** – търси се човек с такива име и се извежда резултат във формат "**{име} -> {номер}**". В случай на несъществуващ контакт, изведете "**Contact {име} does not exist.**".
* **END** – спира получаването на команди.

### Пример

|  |  |
| --- | --- |
| **Вход** | **Изход** |
| A Nakov 0888080808  S Mariika  S Nakov  END | Contact Mariika does not exist.  Nakov -> 0888080808 |
| A Nakov +359888001122  A RoYaL(Ivan) 666  A Gero 5559393  A Simo 02/987665544  S Simo  S simo  S RoYaL  S RoYaL(Ivan)  END | Simo -> 02/987665544  Contact simo does not exist.  Contact RoYaL does not exist.  RoYaL(Ivan) -> 666 |
| A Misho +359883123  A Misho 02/3123  S Misho  END | Misho -> 02/3123 |

### Подсказки

* **Обработвайте командите** като ги разделяте чрез интервал. Изпълнявайте командите докато не стигнете до “**END**”.
* Съхранявайте указателя в **Dictionary<string, string>** с ключ **{име}** и стойност **{телефонен номер}**.

## Подобрен телефонен указател

Добавете следната функционалност към указателя от предната задача **"Телефонен указател":** да **извежда всички контакти в азбучен ред,** когато получи командата “ListAll”.

### Примери

|  |  |
| --- | --- |
| **Вход** | **Изход** |
| A Nakov +359888001122  A RoYaL(Ivan) 666  A Gero 5559393  A Simo 02/987665544  ListAll  END | Gero -> 5559393  Nakov -> +359888001122  RoYaL(Ivan) -> 666  Simo -> 02/987665544 |

### Подсказки

1. **Начин I (по-бавен):** Сортирайте всички записи в речника по ключ и ги изведете.
2. **Начин II (по-бърз):** Пазете записите в по-подходяща структура, която да ги пази в сортиран ред по начало.

## Миньорска задача

Получавате поредица от низове, всеки на нов ред. Всеки нечетен ред на конзолата показва полезно изкопаемо (злато, сребро, мед и т.н.), а всеки четен - количество. Вашата задача е да съберете изкопаемите и да отпечатайте всяко на нов ред.

**Изведете изкопаемите и техните количества във формат:**

**{resource} –> {quantity}**

Количествата ще бъдат в интервала [1 … 2 000 000 000]

### Примери

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Input** | **Output** |  | **Input** | **Output** |
| Gold  155  Silver  10  Copper  17  stop | Gold -> 155  Silver -> 10  Copper -> 17 |  | gold  155  silver  10  copper  17  gold  15  stop | gold -> 170  silver -> 10  copper -> 17 |

## Супермаркет

Напишете програма, която пази информация за продукти и техните цени. Всеки продукт си има име, цена и количество. Ако продуктът не съществува в базата данни, той се добавя със стартово количество.

Ако получите продукт, който вече съществува, то ще увеличите неговото количество и ако цената е различна, ще замените старата с новата цена.

Ще получите име, цена и количество за всеки продукт на нов ред. Накрая ще стои команда “stocked”. При нейното срещане, изведете всичките артикули с техните име, цена, наличност и обща цена на всички продукти с това име. Когато изведете всички продукти, изведете и общата цена на всички артикули.

*Важно: Общата цена се изчислява на базата на най-новата цена за всеки продукт*

### Вход

* Докато не получите “stocked”, ще получавате продукти във формат: “{име} {цена} {количество}”.
* Данните ще бъдат винаги отделени от един интервал

### Изход

* Изведете информацията за всеки продукт, следвайки формата:   
  “{име}: ${цена:F2} \* {количество} = ${общо:F2}”
* На следващия ред, изведете **30 тиренца**.
* На последния ред, изведете общо:  
  “Grand Total: ${grandTotal:F2}”

### Примери

|  |  |
| --- | --- |
| **Вход** | **Изход** |
| Beer 2.20 100  IceTea 1.50 50  NukaCola 3.30 80  Water 1.00 500  stocked | Beer: $2.20 \* 100 = $220.00  IceTea: $1.50 \* 50 = $75.00  NukaCola: $3.30 \* 80 = $264.00  Water: $1.00 \* 500 = $500.00  ------------------------------  Grand Total: $1059.00 |
| Beer 2.40 350  Water 1.25 200  IceTea 5.20 100  Beer 1.20 200  IceTea 0.50 120  stocked | Beer: $1.20 \* 550 = $660.00 Water: $1.25 \* 200 = $250.00 IceTea: $0.50 \* 220 = $110.00 ------------------------------ Grand Total: $1020.00 |
| CesarSalad 10.20 25  SuperEnergy 0.80 400  EvenSupererEnergy 1.00 400  Beer 1.35 350  beer 0.50 450  IceCream 1.50 25  stocked | CesarSalad: $10.20 \* 25 = $255.00  SuperEnergy: $0.80 \* 400 = $320.00  EvenSupererEnergy: $1.00 \* 400 = $400.00  Beer: $1.35 \* 350 = $472.50  beer: $0.50 \* 450 = $225.00  IceCream: $1.50 \* 25 = $37.50  ------------------------------  Grand Total: $1710.00 |