# Lab: Dictionaries, Lambda and LINQ

Test your tasks in the Judge system: <https://judge.softuni.org/Contests/4472>

## Count Real Numbers

Write a program that:

* Read a **list of integers**
* **Print them in ascending order,** along with their **number of occurrences** in the format:

**{number} -> {occurances}**

### Examples

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Input** | **Output** |  | **Input** | **Output** |  | **Input** | **Output** |
| 8 2 2 8 2 | 2 -> 3  8 -> 2 | 1 5 1 3 | 1 -> 2  3 -> 1  5 -> 1 | -2 0 0 2 | -2 -> 1  0 -> 2  2 -> 1 |

## Odd Occurrences

Write a program that extracts all elements from a given sequence of words that are present in it an **odd number of times** (case-insensitive):

* Words are given on a single line, space-separated.
* Print the result elements in lowercase, in their order of appearance.

### Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| 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 |

## Word Synonyms

Create a program, which keeps a dictionary with synonyms. The **key** of the dictionary will be the **word**. The **value** will be a **list of all the synonyms of that word**. You will be given a number **n – the count of the words**. After each word, you will be given a synonym, so the count of lines you have to read from the console is **2 \* n. You will be receiving** a **word** and a **synonym** each on a separate line like this:

* {**word**}
* {**synonym**}

If you get the same word twice, just add the new synonym to the list.

Print the words in the following format:

**"{word} - {synonym1, synonym2, …, synonymN}"**

### Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| 3  cute  adorable  cute  charming  smart  clever | cute - adorable, charming  smart - clever |
| 2  task  problem  task  assignment | task – problem, assignment |

## Word Filter

Write a program that:

* Read an **array of strings**
* Take only words, whose **length is an even number**
* Print **each word on a new line**

### Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| kiwi orange banana apple | kiwi  orange  banana |
| pizza cake pasta chips | cake |