# Exercises: Sets and Dictionaries Advanced

Problems for exercises and homework for the ["C# Advanced" course @ SoftUni](https://softuni.bg/courses/csharp-advanced).

You can check your solutions here: <https://judge.softuni.bg/Contests/1182/>

## Unique Usernames

On the first line you will be given an integer **N**. On the next **N** lines you will receive one username per line. Write a simple program that reads from the console a sequence of **N** usernames and keeps a collection only of the unique ones. Print the collection on the console in order of insertion:

### Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| 6  Ivan  Ivan  Ivan  SemoMastikata  Ivan  Hubaviq1234 | Ivan  SemoMastikata  Hubaviq1234 |

## Sets of Elements

On the first line you are given the length of two sets **n** and **m**. On the next **n** + **m** lines there are **n** numbers that are in the first set and **m** numbers that are in the second one. Find all non-repeating elements that appear in both of them and print those from the **n** set.

Set with length n = 4: {1, **3**, **5**, 7}

Set with length m = 3: {**3**, 4, **5**}

Set that contains all repeating elements -> {**3**, **5**}

### Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| 4 3  1  3  5  7  3  4  5 | 3 5 |
| 2 2  1  3  1  5 | 1 |

## Periodic Table

You are given an **n** number of chemical compounds joined with space (' '). You need to keep track of all chemical elements used in the compounds and at the end print all unique ones in ascending order:

### Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| 4  Ce O  Mo O Ce  Ee  Mo | Ce Ee Mo O |
| 3  Ge Ch O Ne  Nb Mo Tc  O Ne | Ch Ge Mo Nb Ne O Tc |

## Even Times

You are given a list of **N** integer numbers all but one of which appears an odd number of times. Write a program to find the one integer which appears an **even number of times**.

### Examples

|  |  |  |  |
| --- | --- | --- | --- |
| **Input** | **Output** | **Input** | **Output** |
| 3  2  -1  2 | 2 | 5  1  2  3  1  5 | 1 |

## Count Symbols

Write a program that reads some text from the console and counts the occurrences of each character in it. Print the results in **alphabetical** (lexicographical) order.

### Examples

|  |  |  |  |
| --- | --- | --- | --- |
| **Input** | **Output** | **Input** | **Output** |
| SoftUni rocks | : 1 time/s  S: 1 time/s  U: 1 time/s  c: 1 time/s  f: 1 time/s  i: 1 time/s  k: 1 time/s  n: 1 time/s  o: 2 time/s  r: 1 time/s  s: 1 time/s  t: 1 time/s | Did you know Math.Round rounds to the nearest even integer? | : 9 time/s  .: 1 time/s  ?: 1 time/s  D: 1 time/s  M: 1 time/s  R: 1 time/s  a: 2 time/s  d: 3 time/s  e: 7 time/s  g: 1 time/s  h: 2 time/s  i: 2 time/s  k: 1 time/s  n: 6 time/s  o: 5 time/s  r: 3 time/s  s: 2 time/s  t: 5 time/s  u: 3 time/s  v: 1 time/s  w: 1 time/s  y: 1 time/s |

## Wardrobe

You just bought a new wardrobe. The weird thing about it is that it came prepackaged with a big box of clothes (just like those refrigerators, which ship with free beer inside them)! So, you’ll need to find something to wear.

### Input

On the first line of the input, you will receive n – the **number of lines** of clothes, which came prepackaged for the wardrobe.

On the next n lines, you will receive the clothes for each color in the format:

* “{color} -> {item1},{item2},{item3}…”

If a color is added a **second** time, **add** **all items** from it and **count** the **duplicates**.

**Finally**, you will receive the **color** and **item** of the clothing, that you need to look for.

### Output

Go through all the **colors** of the clothes and print them in the following format:

|  |
| --- |
| **{color}** clothes:  \* **{item1}** - **{count}**  \* **{item2}** - **{count}**  \* **{item3}** - **{count}**  …  \* **{itemN}** - **{count}** |

If the **color** lines up with the **clothing item**, print “(found!)” alongside the item. See the examples to better understand the output.

### Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| 4  Blue -> dress,jeans,hat  Gold -> dress,t-shirt,boxers  White -> briefs,tanktop  Blue -> gloves  Blue dress | Blue clothes:  \* dress - 1 (found!)  \* jeans - 1  \* hat - 1  \* gloves - 1  Gold clothes:  \* dress - 1  \* t-shirt - 1  \* boxers - 1  White clothes:  \* briefs - 1  \* tanktop - 1 |

|  |  |
| --- | --- |
| **Input** | **Output** |
| 4  Red -> hat  Red -> dress,t-shirt,boxers  White -> briefs,tanktop  Blue -> gloves  White tanktop | Red clothes:  \* hat - 1  \* dress - 1  \* t-shirt - 1  \* boxers - 1  White clothes:  \* briefs - 1  \* tanktop - 1 (found!)  Blue clothes:  \* gloves - 1 |

|  |  |
| --- | --- |
| **Input** | **Output** |
| 5  Blue -> shoes  Blue -> shoes,shoes,shoes  Blue -> shoes,shoes  Blue -> shoes  Blue -> shoes,shoes  Red tanktop | Blue clothes:  \* shoes - 9 |