# Sets and Dictionaries Advanced

### Count Same Values in Array

Create a program that counts in a given array of double values the number of occurrences of each value.

#### Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| -2.5 4 3 -2.5 -5.5 4 3 3 -2.5 3 | -2.5 - 3 times  4 - 2 times  3 - 4 times  -5.5 - 1 times |
| 2 4 4 5 5 2 3 3 4 4 3 3 4 3 5 3 2 5 4 3 | 2 - 3 times  4 - 6 times  5 - 4 times   1. - 7 times |

### Average Student Grades

Create a program, which reads a **name** of a student and his/her **grades** and **adds** them to the **student record**, then **prints** **the** students' **names** with their **grades** and their **average grade**.

#### Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| 7  John 5.20  Maria 5.50  John 3.20  Maria 2.50  Sam 2.00  Maria 3.46  Sam 3.00 | John -> 5.20 3.20 (avg: 4.20)  Maria -> 5.50 2.50 3.46 (avg: 3.82)  Sam -> 2.00 3.00 (avg: 2.50) |
| 4  Vlad 4.50  Peter 3.00  Vlad 5.00  Peter 3.66 | Vlad -> 4.50 5.00 (avg: 4.75)  Peter -> 3.00 3.66 (avg: 3.33) |
| 5  George 6.00  George 5.50  George 6.00  John 4.40  Peter 3.30 | George -> 6.00 5.50 6.00 (avg: 5.83)  John -> 4.40 (avg: 4.40)  Peter -> 3.30 (avg: 3.30) |

### Product Shop

Create a program that prints information about **food shops** in Sofia and the **products** they **store**. Until the "Revision" command is received, you will be receiving input in the format: "{shop}, {product}, {price}". Keep in mind that if you receive a **shop** you already **have** **received**, you must **collect** its **product** **information**.

Your output must be **ordered** by shop **name** and must be in the format:

"{shop}->

Product: {product}, Price: {price}"

Note: The price should not be rounded or formatted.

#### Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| lidl, juice, 2.30  fantastico, apple, 1.20  kaufland, banana, 1.10  fantastico, grape, 2.20  Revision | fantastico->  Product: apple, Price: 1.2  Product: grape, Price: 2.2  kaufland->  Product: banana, Price: 1.1  lidl->  Product: juice, Price: 2.3 |
| tmarket, peanuts, 2.20  GoGrill, meatballs, 3.30  GoGrill, HotDog, 1.40  tmarket, sweets, 2.20  Revision | GoGrill->  Product: meatballs, Price: 3.3  Product: HotDog, Price: 1.4  tmarket->  Product: peanuts, Price: 2.2  Product: sweets, Price: 2.2 |

### Cities by Continent and Country

Create a program that reads **continents**, **countries,** and their **cities** put them in a **nested dictionary** and **prints** them.

#### Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| 9  Europe Bulgaria Sofia  Asia China Beijing  Asia Japan Tokyo  Europe Poland Warsaw  Europe Germany Berlin  Europe Poland Poznan  Europe Bulgaria Plovdiv  Africa Nigeria Abuja  Asia China Shanghai | Europe:  Bulgaria -> Sofia, Plovdiv  Poland -> Warsaw, Poznan  Germany -> Berlin  Asia:  China -> Beijing, Shanghai  Japan -> Tokyo  Africa:  Nigeria -> Abuja |
| 3  Europe Germany Berlin  Europe Bulgaria Varna  Africa Egypt Cairo | Europe:  Germany -> Berlin  Bulgaria -> Varna  Africa:  Egypt -> Cairo |
| 8  Africa Somalia Mogadishu  Asia India Mumbai  Asia India Delhi  Europe France Paris  Asia India Nagpur  Europe Germany Hamburg  Europe Poland Gdansk  Europe Germany Danzig | Africa:  Somalia -> Mogadishu  Asia:  India -> Mumbai, Delhi, Nagpur  Europe:  France -> Paris  Germany -> Hamburg, Danzig  Poland -> Gdansk |

### Record Unique Names

Create a program, which will take a list of **names** and print **only** the **unique** names in the list.

#### Examples

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Input** | **Output** |  | **Input** | **Output** |  | **Input** | **Output** |
| 8  John  Alex  John  Sam  Alex  Alice  Peter  Alex | John  Alex  Sam  Alice  Peter |  | 7  Lyle  Bruce  Alice  Easton  Shawn  Alice  Shawn  Peter | Lyle  Bruce  Alice  Easton  Shawn |  | 6  Roki  Roki  Roki Roki  Roki  Roki | Roki |

### Parking Lot

Create a program that:

* Records a **car number** for every car that enters the **parking lot**
* Removes a **car number** when the car leaves the **parking lot**

The input will be a string in the format: **"direction, carNumber"**. You will be receiving commands until the **"END"** commandisgiven.

Print the car numbers of the cars, which are still in the parking lot:

#### Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| IN, CA2844AA  IN, CA1234TA  OUT, CA2844AA  IN, CA9999TT  IN, CA2866HI  OUT, CA1234TA  IN, CA2844AA  OUT, CA2866HI  IN, CA9876HH  IN, CA2822UU  END | CA9999TT  CA2844AA  CA9876HH  CA2822UU |
| IN, CA2844AA  IN, CA1234TA  OUT, CA2844AA  OUT, CA1234TA  END | Parking Lot is Empty |

### SoftUni Party

#### There is a party in SoftUni. Many guests are invited and there are two types of them: VIP and Regular. When a guest comes, check if he/she exists in any of the two reservation lists.

All reservation numbers will be with the length of 8 chars.

All VIP numbers start with a digit.

First, you will be receiving the reservation numbers of the guests. You can also receive 2 possible commands:

* "PARTY" – After this command, you will begin receiving the reservation numbers of the people, who came to the party.
* "END" – The party is over and you have to stop the program and print the appropriate output.

In the end, print the count of the guests who didn't come to the party, and afterward, print their reservation numbers. the VIP guests must be first.

#### Examples

|  |  |  |  |
| --- | --- | --- | --- |
| **Input** | **Output** | **Input** | **Output** |
| 7IK9Yo0h  9NoBUajQ  Ce8vwPmE  SVQXQCbc  tSzE5t0p  PARTY  9NoBUajQ  Ce8vwPmE  SVQXQCbc  END | 2  7IK9Yo0h  tSzE5t0p | m8rfQBvl  fc1oZCE0  UgffRkOn  7ugX7bm0  9CQBGUeJ  2FQZT3uC  dziNz78I  mdSGyQCJ  LjcVpmDL  fPXNHpm1  HTTbwRmM  B5yTkMQi  8N0FThqG  xys2FYzn  MDzcM9ZK  PARTY  2FQZT3uC  dziNz78I  mdSGyQCJ  LjcVpmDL  fPXNHpm1  HTTbwRmM  B5yTkMQi  8N0FThqG  m8rfQBvl  fc1oZCE0  UgffRkOn  7ugX7bm0  9CQBGUeJ  END | 2  xys2FYzn  MDzcM9ZK |

## Unique Usernames

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

### Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| 6  John  John  John  Peter  John  Boy1234 | John  Peter  Boy1234 |
| 10  Peter  Maria  Peter  George  Sam  Maria  Sara  Peter  Sam  George | Peter  Maria  George  Sam  Sara |

## Sets of Elements

Create a program that prints a **set of elements**. On the first line, you will receive two numbers - **n** and **m**, which represent the lengths of two separate sets. On the next **n** + **m** lines, you will receive **n** numbers, which are the numbers in the **first** set, and **m** numbers, which are in the **second** set. Find all the **unique** **elements** that appear in **both of them** and **print** them in the order in which they appear in the **first** set - **n**.

**For example:**

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

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

Set that contains all the **elements** that repeat in **both** **sets** -> {**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

Create a program that keeps all the **unique** chemical **elements**. On the first line, you will be given a number **n** - the **count** of input **lines** that you are going to receive. On the next **n** lines, you will be receiving **chemical** **compounds**, separated by a **single** **space**. Your task is to print all the **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

Create a program that **prints** a **number** from a collection, which appears an **even** **number** of **times** in it. On the first line, you will be given **n** – the **count** of **integers** you will receive. On the next n lines, you will be receiving **the numbers**. It is **guaranteed** that **only** **one** of them **appears** an **even** **number** of times. Your task is to **find** that **number** and **print** it in the end.

### Examples

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

## Count Symbols

Create 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** |
| 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 |