# Exercises: Lists and Matrices

Problems for exercises and homework for the [“Programming Fundamentals” course @ SoftUni](https://softuni.bg/courses/programming-fundamentals).

You can check your solutions here: <https://judge.softuni.bg/Contests/173/Lists-and-Matrices-Lab>.

## Remove Negatives and Reverse

Read a **list of integers**, **remove all negative numbers** from it and print the remaining elements in **reversed order**. In case of no elements left in the list, print “empty”.

### Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| 10 -5 7 9 -33 50 | 50 9 7 10 |
| 7 -2 -10 1 | 1 7 |
| -1 -2 -3 | empty |

### Hints

* Read a text line from the console, split it by space, parse the obtained items as integers and convert them to list of integers.
* Create a new empty list for the results.
* Scan the input list from the end to the beginning. Check each element and append all non-negative elements to the result list.
* Finally, print the results list (at a single line holding space-separated numbers).

## Append Lists

Write a program to **append several lists** of numbers.

* Lists are separated by ‘|’.
* Values are separated by spaces (‘ ’, one or several)
* Order the lists from the **last** to the **first**, and their values from **left** to **right**.

### Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| 1 2 3 |4 5 6 | 7 8 | 7 8 4 5 6 1 2 3 |
| 7 | 4 5|1 0| 2 5 |3 | 3 2 5 1 0 4 5 7 |
| 1| 4 5 6 7 | 8 9 | 8 9 4 5 6 7 1 |

### Hints

* Create a new empty list for the results.
* Split the input by ‘|’ into array of tokens.
* Pass through each of the obtained tokens from tight to left.
  + For each token, split it by space and append all non-empty tokens to the results.
* Print the results.

## Sum Adjacent Equal Numbers

Write a program to **sum all adjacent equal numbers** in a list of decimal numbers, starting from **left to right**.

* After two numbers are summed, the obtained result could be equal to some of its neighbors and should be summed as well (see the examples below).
* Always sum the **leftmost** two equal neighbors (if several couples of equal neighbors are available).

### Examples

|  |  |  |
| --- | --- | --- |
| **Input** | **Output** | **Explanation** |
| 3 3 6 1 | 12 1 | **3 3** 6 1 🡪 **6 6** 1 🡪 12 1 |
| 8 2 2 4 8 16 | 16 8 16 | 8 **2 2** 4 8 16 🡪 8 **4 4** 8 16 🡪 **8 8** 8 16 🡪 16 8 16 |
| 5 4 2 1 1 4 | 5 8 4 | 5 4 2 **1 1** 4 🡪 5 4 **2 2** 4 🡪 5 **4 4** 4 🡪 5 8 4 |

### Hints

1. Read the **input** and parse it to **list of numbers**.
2. Find the **leftmost** two **adjacent equal cells**.
3. **Replace** them with their **sum**.
4. **Repeat** (1) and (2) until no two equal adjacent cells survive.
5. **Print** the processed list of numbers.

## Split by Word Casing

Read a **text**, split it into words and distribute them into **3 lists**.

* **Lower-case words** like “programming”, “at” and “databases” – consist of lowercase letters only.
* **Upper-case words** like “PHP”, “JS” and “SQL” – consist of uppercase letters only.
* **Mixed-case words** like “C#”, “SoftUni” and “Java” – all others.

Use the following **separators** between the words: , ; : . ! ( ) " ' \ / [ ] space

Print the 3 lists as shown in the example below.

### Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| Learn programming at SoftUni: Java, PHP, JS, HTML 5, CSS, Web, C#, SQL, databases, AJAX, etc. | Lower-case: programming, at, databases, etc  Mixed-case: Learn, SoftUni, Java, 5, Web, C#  Upper-case: PHP, JS, HTML, CSS, SQL, AJAX |

### Hints

* **Split** the input text using the above described **separators**.
* **Process** the obtained **list of words** one by one.
* Create 3 lists of words (initially empty): lowercase words, mixed-case words and uppercase words.
* Check each word and append it to one of the above 3 lists:
  + Count the **lowercase letters** and **uppercase letters**.
  + If all letters are **lowercase**, append the word to the lowercase list.
  + If all letters are **uppercase**, append the word to the uppercase list.
  + Otherwise the word is considered mixed-case 🡪 append it to the mixed-case list.
* Print the obtained 3 lists as shown in the example above.

## Sort Numbers

Read a **list of decimal numbers** and **sort** them.

### Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| 8 2 7 3 | 2 <= 3 <= 7 <= 8 |
| 2 4 -9 | -9 <= 2 <= 4 |

## Count Numbers

Read a **list of integers** in range [0…1000] and **print them in ascending order** along with their **number of occurrences**.

### Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| 8 2 2 8 2 2 3 7 | 2 -> 4  3 -> 1  7 -> 1  8 -> 2 |
| 10 8 8 10 10 | 8 -> 2  10 -> 3 |

## Square Numbers

Read a **list of integers** and **extract square numbers** from it and print them in **descending order**. A square number is an integer which is the square of any integer. For example, 1, 4, 9, 16 are square numbers.

### Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| 3 4 5 6 8 9 | 9 4 |
| 1 2 9 4 16 25 49 36 | 49 36 25 16 9 4 1 |

## Build a Matrix of Letters

Build a **matrix** of capital Latin letters of size **rows** x **cols** like at the example below.

### Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| 2  2 | A B  C D |
| 3  7 | A B C D E F G  H I J K L M N  O P Q R S T U |

## Rotate a Matrix

Write a program to read a **matrix of words** (space separated) and **rotate** it on the right as shown in the examples.

### Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| 3  4  A B C D  E F G H  I J K L | I E A  J F B  K G C  L H D |
| 3  3  Hi PHP Java  C# SQL JSON  HTML CSS JS | HTML C# Hi  CSS SQL PHP  JS JSON Java |