# Exercises: Functional Programming

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

Submit your solutions in the SoftUni judge system at <https://judge.softuni.bg/Contests/199/>.

## Action Point

Write a program that reads a collection of **strings** from the console and then **prints** them onto the **console**. Each name should be printed on a **new** **line**. Use **Action<T>**.

### Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| Pesho Gosho Adasha | Pesho  Gosho  Adasha |

## Knights of Honor

Write a program that reads a collection of **names** as **strings** from the **console** then appends “**Sir**” in front of every name and **prints** it back onto the **console**. Use **Action<T>**.

### Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| Pesho Gosho Adasha StanleyRoyce | Sir Pesho  Sir Gosho  Sir Adasha  Sir StanleyRoyce |

## Custom Min Function

Write a simple program that reads from the **console** a set of **integers** and **prints** back onto the **console** the **smallest** **number** from the collection. Use **Func<T, T>**.

### Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| 1 4 3 2 1 7 13 | 1 |

## Find Evens or Odds

You are given a lower and an upper bound for a range of integer numbers. Then a command specifies if you need to list all even or odd numbers in the given range. Use **Predicate<T>**.

### Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| 1 10  odd | 1 3 5 7 9 |
| 20 30  even | 20 22 24 26 28 30 |

## Applied Arithmetics

Write a program that executes some mathematical operations on a given collection. On the **first line** you are given **a list of numbers**. On the **next lines** you are passed **different commands** that you need to **apply to all numbers** in the list: "add" -> add 1 to each number; "multiply" -> multiply each number by 2; "subtract" -> subtract 1 from each number; “print” -> print the collection. The input will end with an "**end**" command. Use functions.

### Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| 1 2 3 4 5  add  add  print  end | 3 4 5 6 7 |
| 5 10  multiply  subtract  print  end | 9 19 |

## Reverse and Exclude

Write a program that reverses a collection and removes elements that are divisible by a given integer **n**. Use predicates/functions.

### Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| 1 2 3 4 5 6  2 | 5 3 1 |
| 20 10 40 30 60 50  3 | 50 40 10 20 |

## Predicate for Names

Write a program that filters a list of names according to their length. On the first line you will be given integer **n** representing name length. On the second line you will be given some names as strings separated by space. Write a function that prints only the names whose length is **less than or equal** to **n**.

### Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| 4  Kurnelia Qnaki Geo Muk Ivan | Geo  Muk  Ivan |
| 4  Karaman Asen Kiril Yordan | Asen |

## Custom Comparator

Write a custom comparator that sorts all even numbers before all odd ones in ascending order. Pass it to an Array.Sort() function and print the result. Use functions.

### Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| 1 2 3 4 5 6 | 2 4 6 1 3 5 |
| -3 2 | 2 -3 |