Programming Fundamentals

P.PORTO

ESCOLA
SUPERIOR
DE TECNOLOGIA
E GESTÃO

Informatics Engineering
Computer and Networks Security

Worksheet 5

Subject(s):

- Functions.
- Standard library.
- Modular programming.

Part 1

1. Implement a program that reads an integer value and creates a function that prints on a line this number of asterisks. The integer value must be read through a function.
2. Implement a program that requests two integers from the user and then allows the user to choose between 4 arithmetic operations: addition, subtraction, multiplication, and division. Each of these operations must be implemented through a specific function for that purpose.
3. Implement and use a module (**`myIntLib`**) with the arithmetic functions of the previous exercise.
4. Implement a function that, given 2 matrices of integers (with the same dimension) as well as their dimension, presents (in a function) in matrix form the result of the sum of the 2 matrices and (in another function) the sum of their elements.
5. Trace the function presented on page 6 of the theoretical slides on recursiveness.

Part 2

1. Change the function of exercise 1 to validate the allowed value range. The function must have two parameters to identify the allowed value range. Add the function to the module (**`myIntLib`**). Test the implementation.
2. Implement a program that reads the notes of a random number of students. Consider that the notes have a value range of 0-20 using the **`myIntLib`** module. Display the average of the grades read.
3. Add to the **`myIntLib`** module two functions that return the maximum and minimum value of two whole values steps as arguments. Test the implementation as read values of the user.
4. Considering the following mathematical definition of the recursive function to find the power of a number. The function accepts two numbers **`x`** and **`y`** and calculates x^y. Add this function to the module.
5. Implement a menu for the functions of the module you have implemented, thus creating a calculator with several functions. The values read must use the function implemented in exercise 1.

Part 3 (optional)

1. 1. It is intended to change the application created to solve the problem in Part 3 of Practice Sheet 4 and to adopt principles of modularity in the code using the implementation of functions and

procedures.
Suggestions:

- Create a function for the resolution of each of the following points: b) to e)
- Create a function that has as parameters a text to be presented to the user, as well as a lower and upper numerical limit and validates that the entered value is within that limit. Use this function when reading the employee code and identifying the number of days worked.