

PRACTICE 2: Input and Output (Classroom Exercises)

DURATION: 2 Hours

THEORIC CONTENTS

- Lesson 2. Algorithms.
- Lesson 3. Data Types.
- Lesson 4. Operators and Expressions.
- Lesson 5. Input and Output.

LEARNING OBJECTIVES

- **LO-02.1.** Identify the Variables that are required in algorithm design.
- **LO-02.2.** Identify the Constants that are required in algorithm design.
- **LO-03.1.** Choose correctly the data type of a variable according to the values that it could store in algorithm design.
- **LO-03.2.** Choose correctly the data type of a constant according to the values that it could store in algorithm design.
- **LO-05.1.** Implement variables in C language in function of the type and range of values that is needed in the design of a given algorithm.
- **LO-05.2.** Implement constants in C language in function of the type and values that is needed in the design of a given algorithm.
- **LO-10.1.** Define and Evaluate expressions, considering values, variables, constants and precedence rules and order of evaluation of each operator.
- **LO-11.1.** Implement in C Language expressions using arithmetic operators.
- **LO-11.2.** Implement in C Language expressions using assignment operator.
- **LO-14.1.** Identify standard input operations in the algorithm design.
- **LO-14.2.** Identify standard output operations in the algorithm design.
- **LO-14.3.** Identify text file input operations in the algorithm design.
- **LO-14.4.** Identify text file output operations in the algorithm design
- **LO-15.1.** Implement standard input operations in C language to collect data through variables.
- **LO-15.2.** Implement standard output operations in C language to show data through variables.
- **LO-15.3.** Implement text file input operations in C language to collect data through variables.
- **LO-15.4.** Implement text file output operations in C language to show data through variables.

GITHUB CLASSROOM ASSIGNMENT

<https://classroom.github.com/a/c3-n2dSB>

PROPOSED EXERCISES

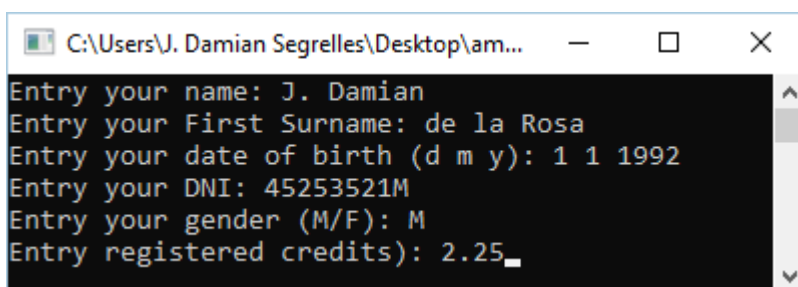
Exercise 1. Design and implement a C program that solves the next problem definition:

The program has to ask the following data of a student:

- Name. It can be a composed name (ex. "Jose Damian").
- First Surname. It can be a composed name (ex. "de la Rosa").
- DNI. Including the number and the letter. (ex. 67859402N).
- Date of birth. It has to be composed by three number which corresponds to the day, month and year respectively. (ex. "01 12 2009").
- Gender. The valid values only are 'M' for Male or 'F' for female.
- Registered Credits. It can contain decimals. (ex. 3.25)

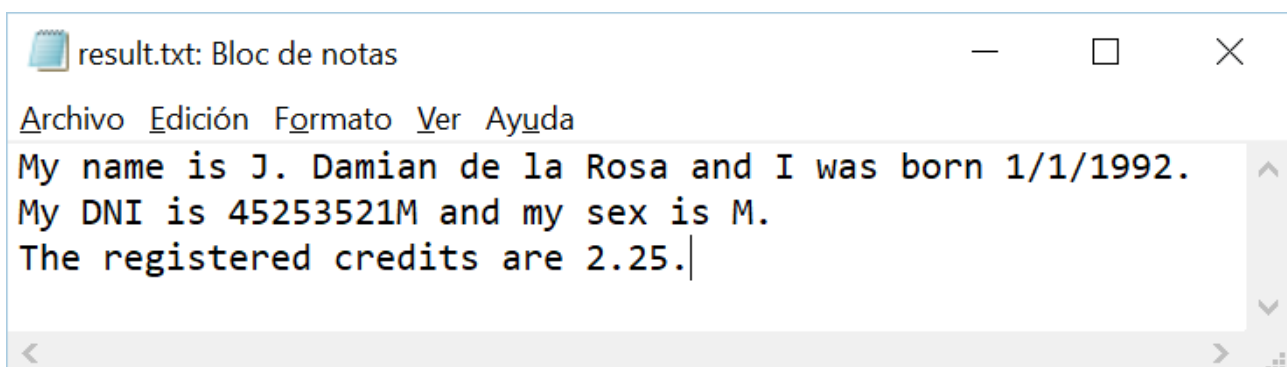
Once all data have been entered, the program should write the student data in the file "result.txt"

Example of Execution:



```
C:\Users\J. Damian Segrelles\Desktop\am...
Entry your name: J. Damian
Entry your First Surname: de la Rosa
Entry your date of birth (d m y): 1 1 1992
Entry your DNI: 45253521M
Entry your gender (M/F): M
Entry registered credits): 2.25_
```

The following figure shows the content the file "result.txt" should have once the program has finished.



```
result.txt: Bloc de notas
Archivo Edición Formato Ver Ayuda
My name is J. Damian de la Rosa and I was born 1/1/1992.
My DNI is 45253521M and my sex is M.
The registered credits are 2.25.
```

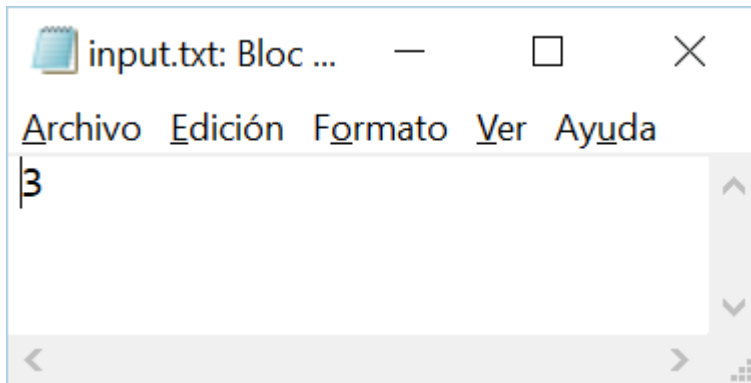
Exercise 2. Design and implement a C program that solves the next problem definition:

Computes the square of an integer number, which must be read from input.txt. The result should be written to the file output.txt.

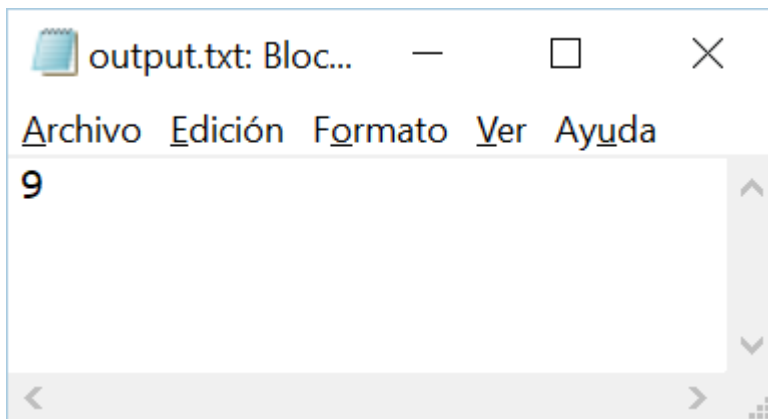
Before to execute the program, you have to create “input.txt” file using Notepad program.

Example of Execution:

Input File (input.txt)



Output File (output.txt)



Exercise 3. Consider the following table as the list of products that are served at a bar, with their corresponding prices.

Sandwich/Roll	Price
Sausage	1.10
Tuna	1.40

Drinks	Price
Water	0.50
Coca Cola	0.75
Orange Juice	0.70

Design and implement a program that asks the user how much has consumed of each product. It is not allowed half parts, only whole units. Once, all the data have been entered, the program should display the total bill corresponding to the user and should receive how much has paid the user.

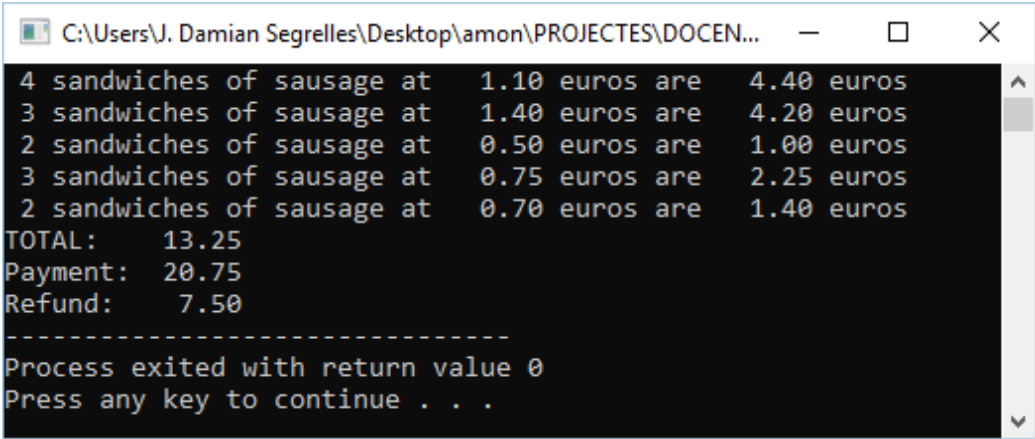
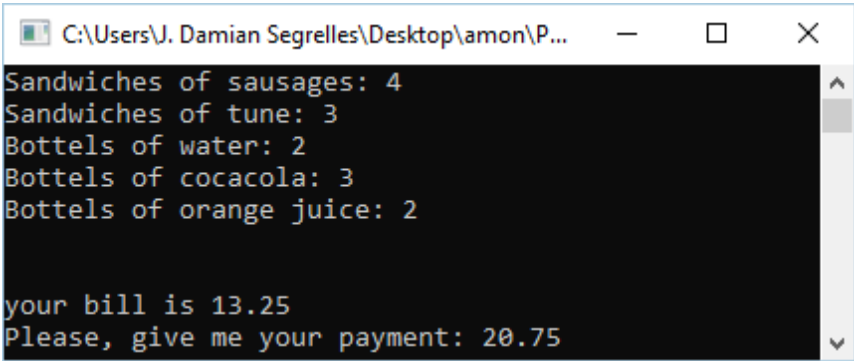
Next, the screen should be cleared with `system("cls")` of `stdlib.h` and the following messages should be displayed:

NOTE: Prices must be declared as fixes values.

```
xx sandwiches of sausage at  xx.xx euros are xxx.xx euros
xx sandwiches of tuna at    xx.xx euros are xxx.xx euros
xx bottles of water at      xx.xx euros are xxx.xx euros
xx bottles of coca cola at  xx.xx euros are xxx.xx euros
xx glasses of orange juice at xx.xx euros are xxx.xx euros

TOTAL:    xxx.xx euros
Payment:  xxx.xx euros
Refund:    xxx.xx euros
```

Example of Execution:



Exercise 4. Design and implement a C program that solves the next problem definition:

A car parts company needs a program that computes and displays the selling price (Euros) of its products. In this sense, the program should apply the following formula:

$$\begin{aligned} \text{profit} &= \text{costprice} * \frac{\text{margin}}{100.0} \\ \text{sellingprice} &= \text{costprice} + \text{profit} \end{aligned}$$

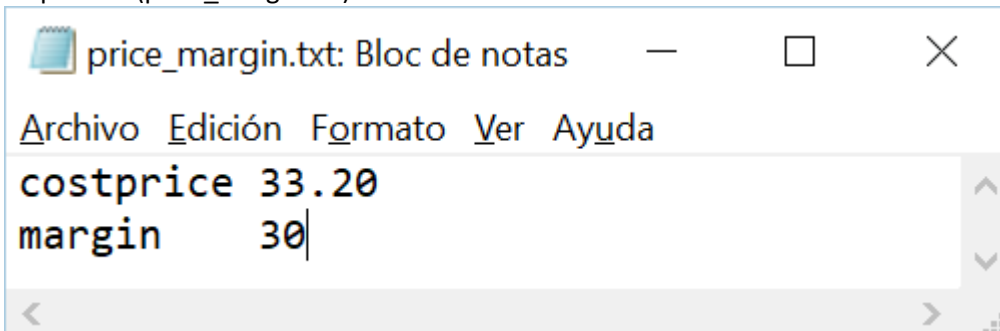
The cost price and the profit margin are specified in a file, (price_margin.txt):
 . The program should read such values and then, it should compute the selling price. The result should be displayed on the screen and added to the input file.

Before to execute the program, you have to create “price_margin.txt” file using Notepad program where it stores the cost Price and the profit margin.

NOTE: The cost price has to be composed by a number that corresponds to euros and cents, and the profit margin the percentage without decimal part.

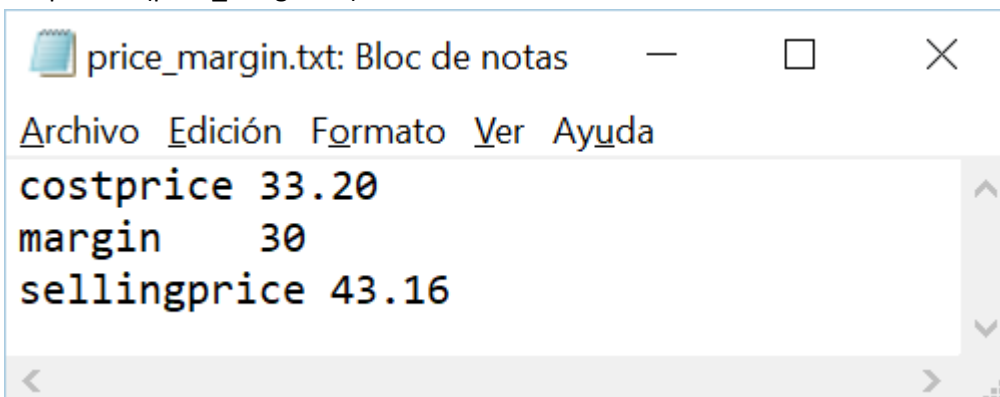
Example of Execution:

Input File (price_margin.txt):



```
price_margin.txt: Bloc de notas
Archivo Edición Formato Ver Ayuda
costprice 33.20
margin 30
```

Output File (price_margin.txt):



```
price_margin.txt: Bloc de notas
Archivo Edición Formato Ver Ayuda
costprice 33.20
margin 30
sellingprice 43.16
```

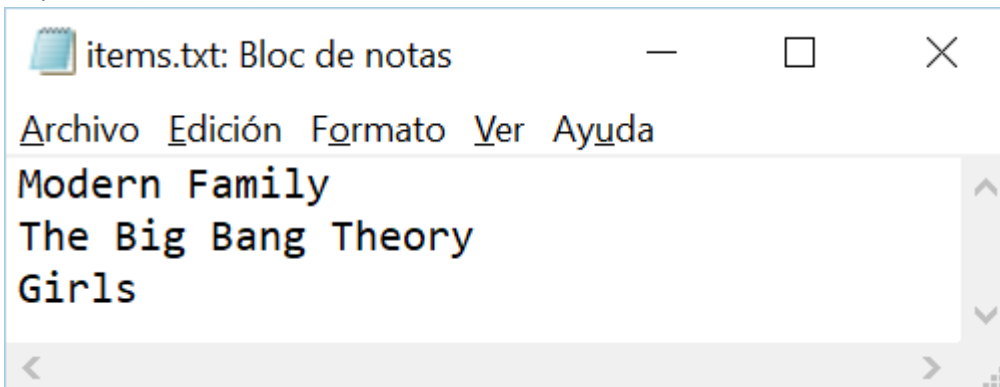
Exercise 5. Design and implement a C program that solves the next problem definition:

The program has to copy each line of the file “items.txt” to “items2.txt”, considering that each line in the destination file must be preceded by the line number. Consider the origin file has 3 lines.

Example:

Example of Execution:

Input File (items.txt):



Output File (items2.txt):

