DURATION: 2 Hours

THERIC CONTENTS

* Lesson 2. Algorithms.
* Lesson 3. Data Types.
* Lesson 4. Operators and Expressions.
* Lesson 5. Input and Output.
* Lesson 6. Flow Control Selection.
* Lesson 7. Flow Control Repetition.
* Lesson 8. Modular Programming

GITHUB CLASSROOM ASSIGNMENT

https://classroom.github.com/a/ovqh3Vwl

PROPOSED EXERCISES

Exercise 1. Design and implement a program in C using modular programing that computes the ex-according to the Tailor series:



Exercise 2. Design and implement a program in C using modular programing that computes and displays the number of ways of selecting ***n*** elements from a set that contains *m* elements (combination, in mathematics).

Combination =

n!

m! \* (n-m)!

Exercise 3. Design and implement a program in C using modular programing that that offers the following features:

1. Copy the content of a file to another file.
2. Display the number of characters in a file. The user should choose whether or not the spaces are included.
3. Display the number of words in a file.
4. Display the number of lines in a file.
5. Given a file, write to the end, a summary that includes the following information:
   1. Number of characters (with spaces).
   2. Number of words.
   3. Number of lines.

 The filenames must be given by the user. Each new feature must be specified as a function of the program.

Exercise 4. Design and implement a program in C using modular programming that displays the following menu:

CALCULATOR

--------------------

1. Exit
2. Add
3. Subtract
4. Multiply
5. Divide
6. Power Of
7. Factorial
8. ex (using Tailor Serie)

The program must read the choice made by the user (0, 1, 2, 3, 4, 5, 6 or 7). If the input is zero, the program has to end. If the input is any other value, then the program has to ask a new choice. Once the result is shown, clear the screen and show the menu again.

Exercise 5.Design and implement a program in C using modular programming in which the file **"marks.txt"** contains the marks that students have obtained in a subject. Considering the content of this file, write a C program that displays the following information:

1. Minimum and Maximum marks**.**
2. Number of students that have passed the subject.
3. Average mark.
4. Number of students that have obtained a mark higher than the mark average.
5. Number of students that have obtained a mark lower than the mark average.

|  |  |
| --- | --- |
|  |  |