**Software Engineering Department**

**Computer Organisation and Programming Course   
final assignment**

**Pocket Calculator application**

**Written by:**

**Coral Rubilar 316392877**

**31 July 2019**

**Lecturer: Dr Yigal Hoffner**

Table des matières

[1 Pocket Calculator application design 3](#_Toc15412107)

[1.1 Extra work carried out 3](#_Toc15412108)

[1.2 Major design/implementation decisions 6](#_Toc15412109)

[1.3 The high-level algorithms 7](#_Toc15412110)

[2 The User Guide: 9](#_Toc15412111)

[3 The Mano Code 11](#_Toc15412112)

# Pocket Calculator application design

Hello and welcome to Mélissa’s and Coral’s pocket calculator. After some hard work and white nights our amazing calculator is ready 😊.

This pocket calculator was programmed in Mano in assembly language, that has been taught throughout the semester.

The calculator performs four basic arithmetical operations which are:

1. addition

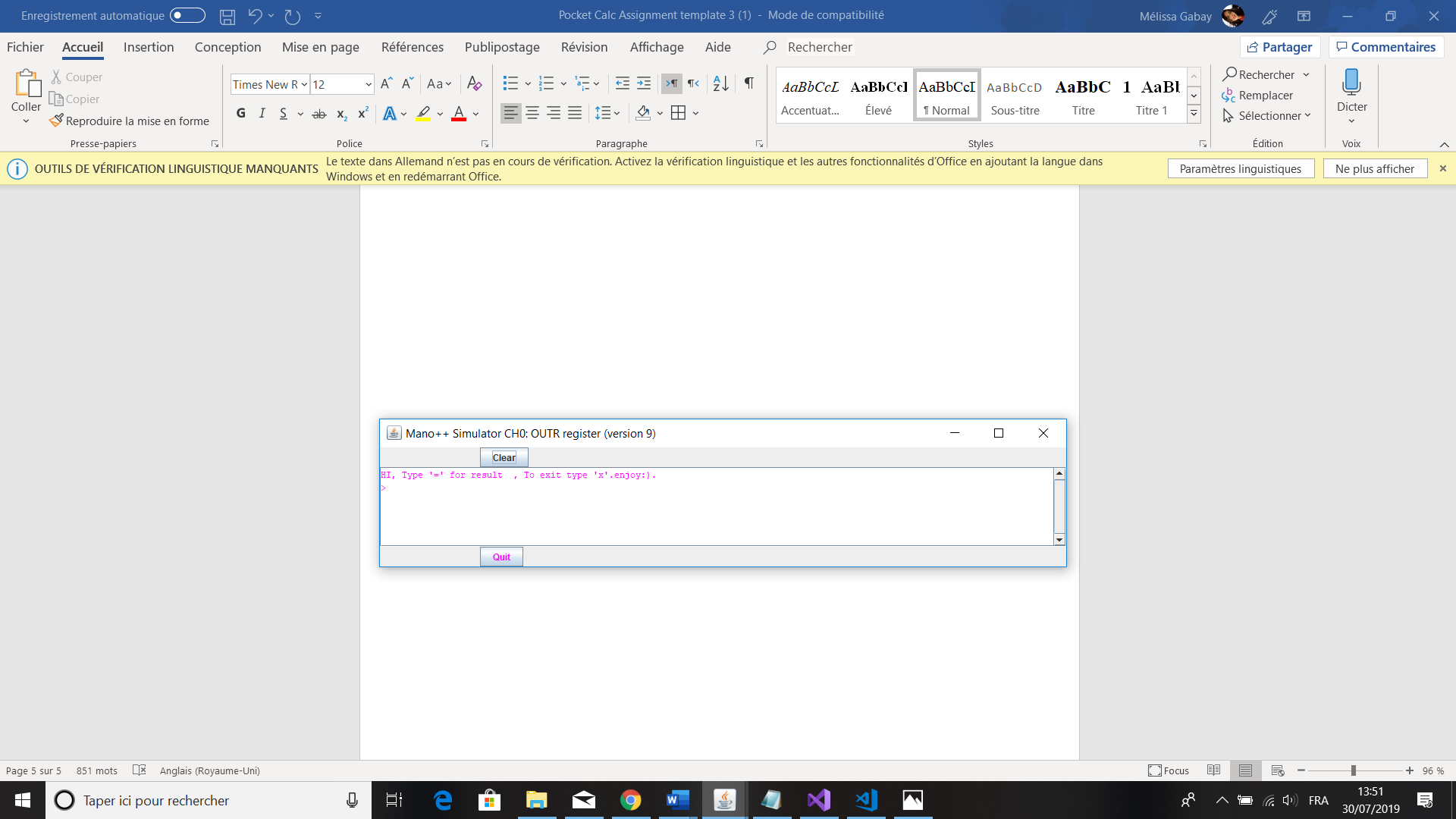
2. subtraction

3. multiplication

4. division

With the introductory sentence “HI, Type '=' for result , To exit type 'x'.enjoy:).” the user understands what he must do, meaning the user should input a true arithmetical operation.

The number must be between - 32,768 and 32,767.



If you enter an incorrect character (like a letter or two arithmetical characters following except ) the computer will print **ERROR** and then prints back the introductory sentence, the user will have to rewrite a true arithmetical operation.

To finish using the calculator, the user must press "x" and "Enter" when it asks for “write a mathematical operation” then the program will print "Bye" and will stop.

## Extra work carried out

To start the user can only enter or a number.

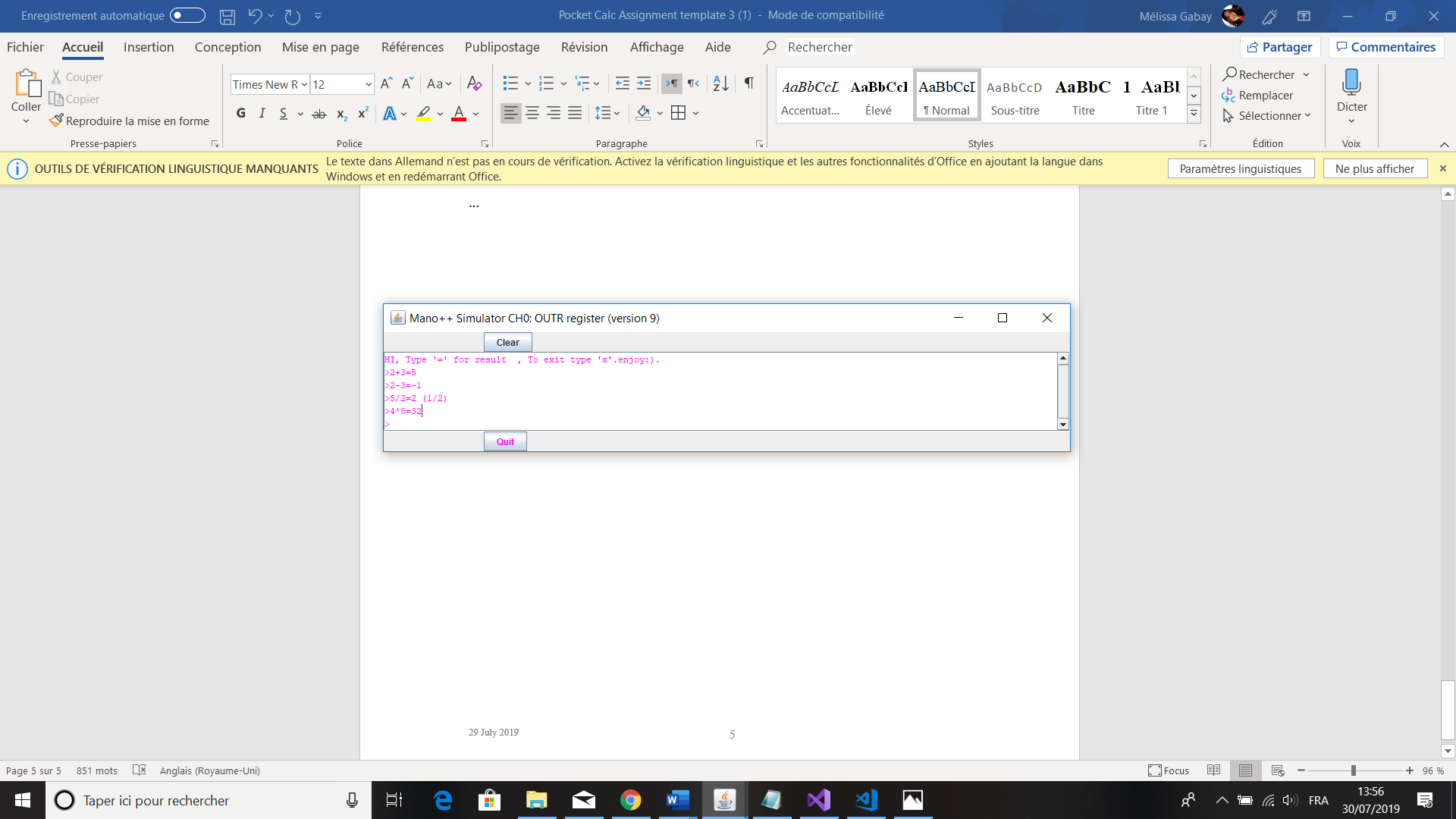
Then the user must enter only an arithmetical character, meaning: , , or of course .

To continue the user can only enter or a number.

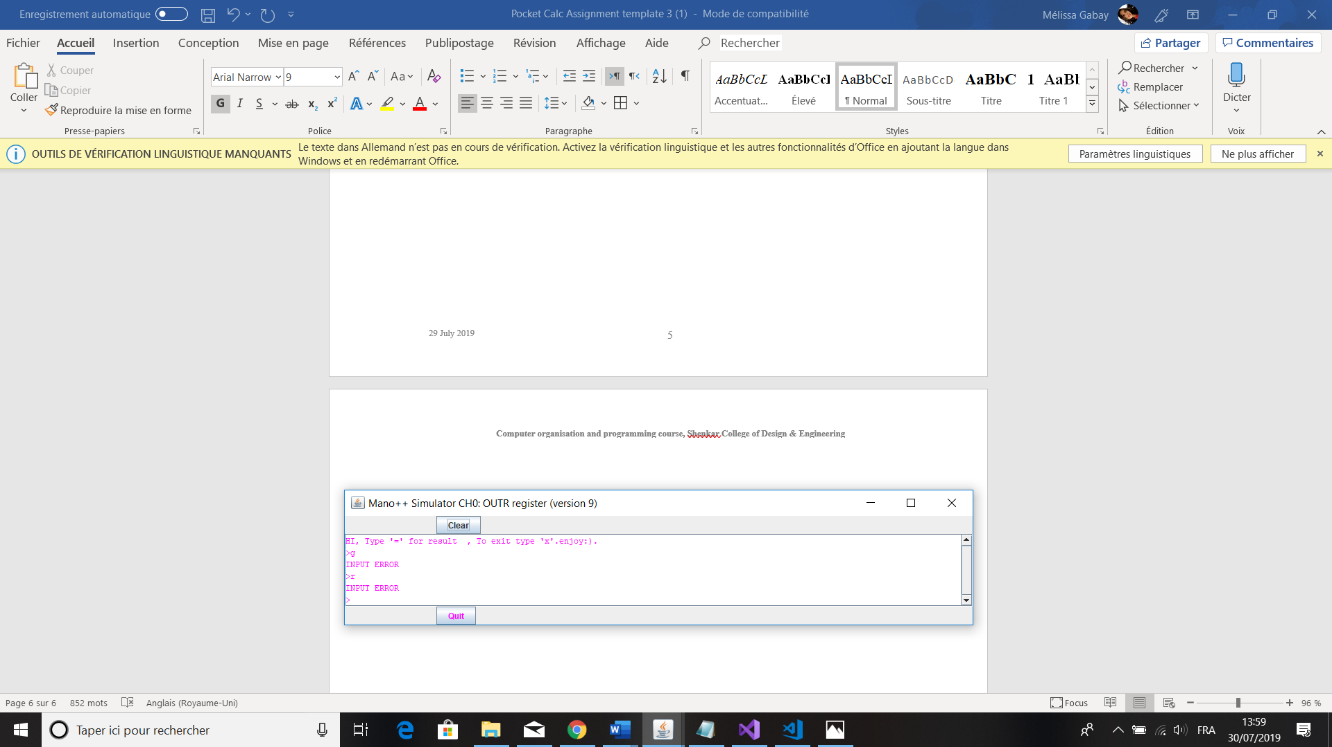
Finally, the user inputs the character “ = ” to end the equation.

Additional points:

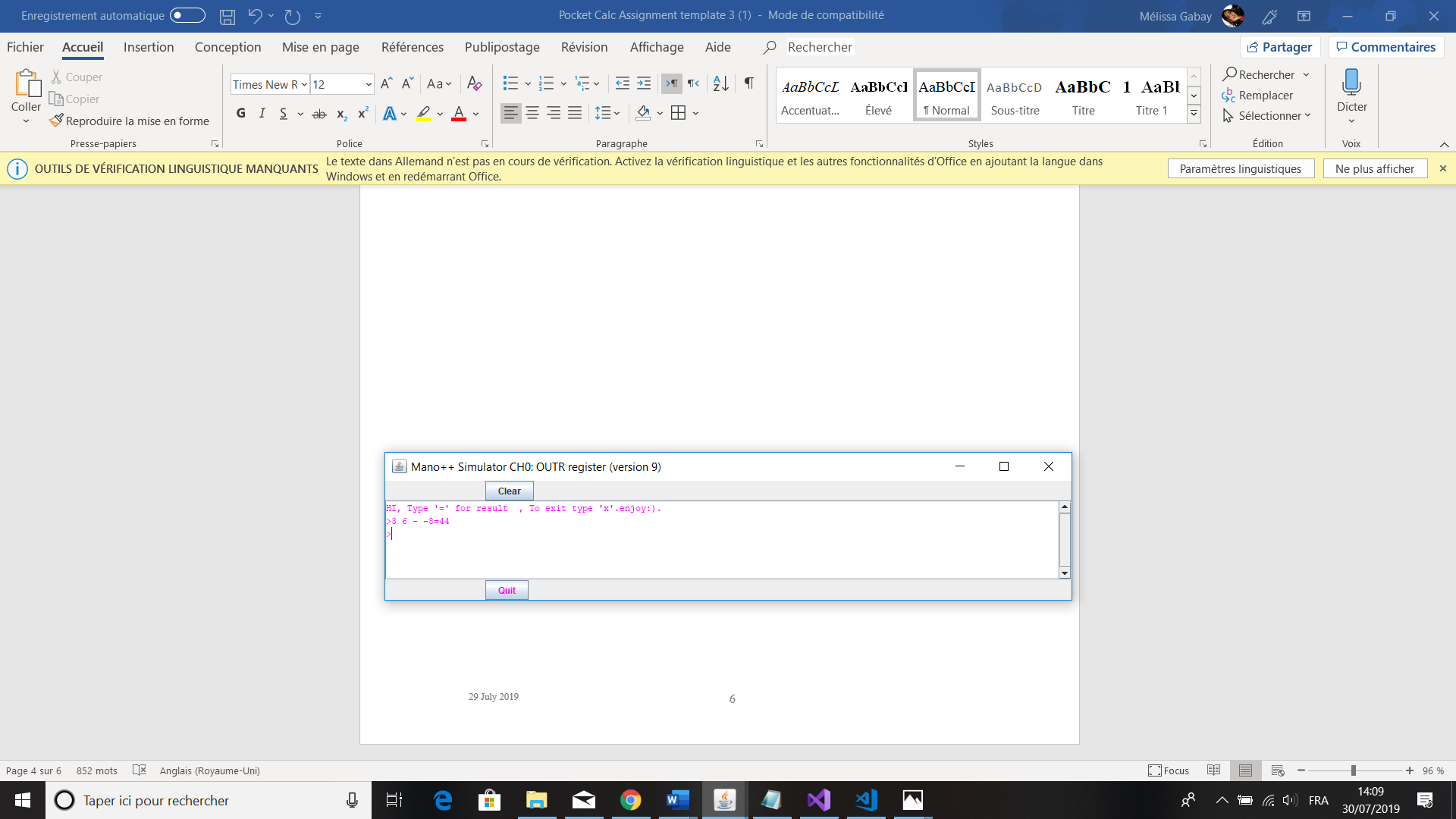
* We used the sophisticated input to do the work meaning that you don’t have to put the operand, the first and the second number you only input a normal equation and you write ” = ” to enter do the operation. Example:
  1. 2 + 3 =
  2. 2 - 3 =
  3. 5 / 2 =
  4. 4 \* 8 =



* You don’t need to put “ = “ if you write an incorrect character the computer understands that it’s not a true character.

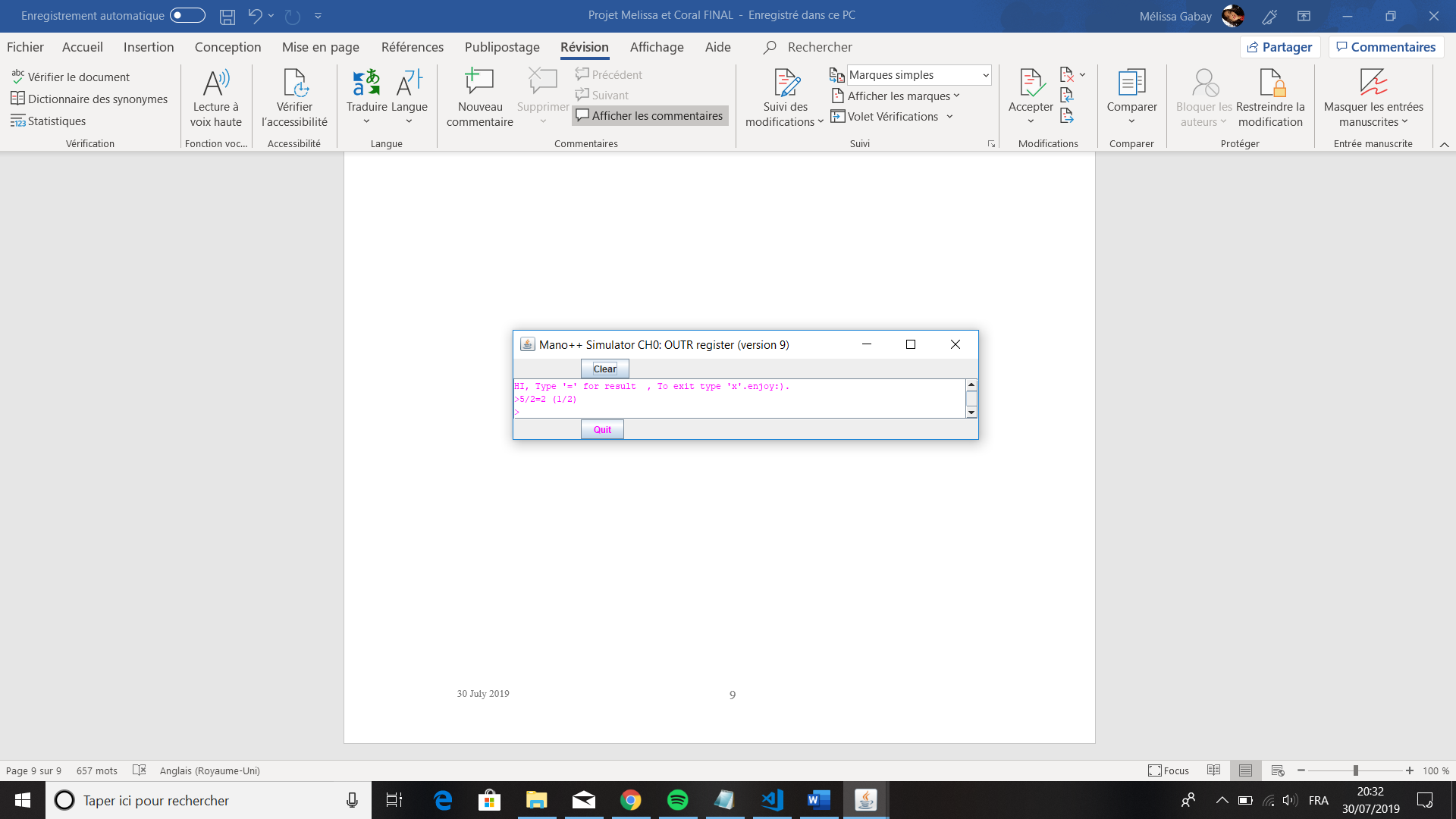


* The computer ignores if you write “space”. Example:
  1. 3 6 - -8= 36 - -8 = 44



## Major design/implementation decisions

* About negatives numbers, the program saves it as a positive number (absolute values) and recognises that it’s negative with the flag of minus numbers passing from 0 to 1.
* When you divide two numbers the remainder will be printed in the form of a fraction.



## The high-level algorithms

int main ()

{

printf ("HI, Type '=' for result , To exit type 'x'.enjoy:).");

int Num1 = 0;

int Num2 = 0;

char op = NULL;

char op\_eq = NULL;

while\_loop, while (1)

{

Num1 = 0;

Num2 = 0;

op = NULL;

op\_eq = NULL;

printf (">");

scanf ("%d%c%d%c", &Num1, &op, &Num2, &op\_eq);

if (Num1 == 'x' || Num2 == 'x' || op == 'x' || op\_eq == 'x')

    THEN printf ("BYE\n");

return 0;

FI;

if (op\_eq != '=')

    THEN printf ("\n INPUT ERROR \n");

goto while\_loop;

FI;

switch (op)

    {

    case '+':

     plus\_f (num1, num2);

     break;

    case '-':

     minus\_f (num1, num2);

     break;

    case '/':

     if (num2 == 0)

     THEN printf ("\n ERROR! \n Can not divide by 0! \n");

     goto while\_loop;

     FI;

     div\_f (num1, num2);

     break;

    case '\*':

     multi\_f (num1, num2);

     break;

    default:

     printf ("\n INPUT ERROR \n");

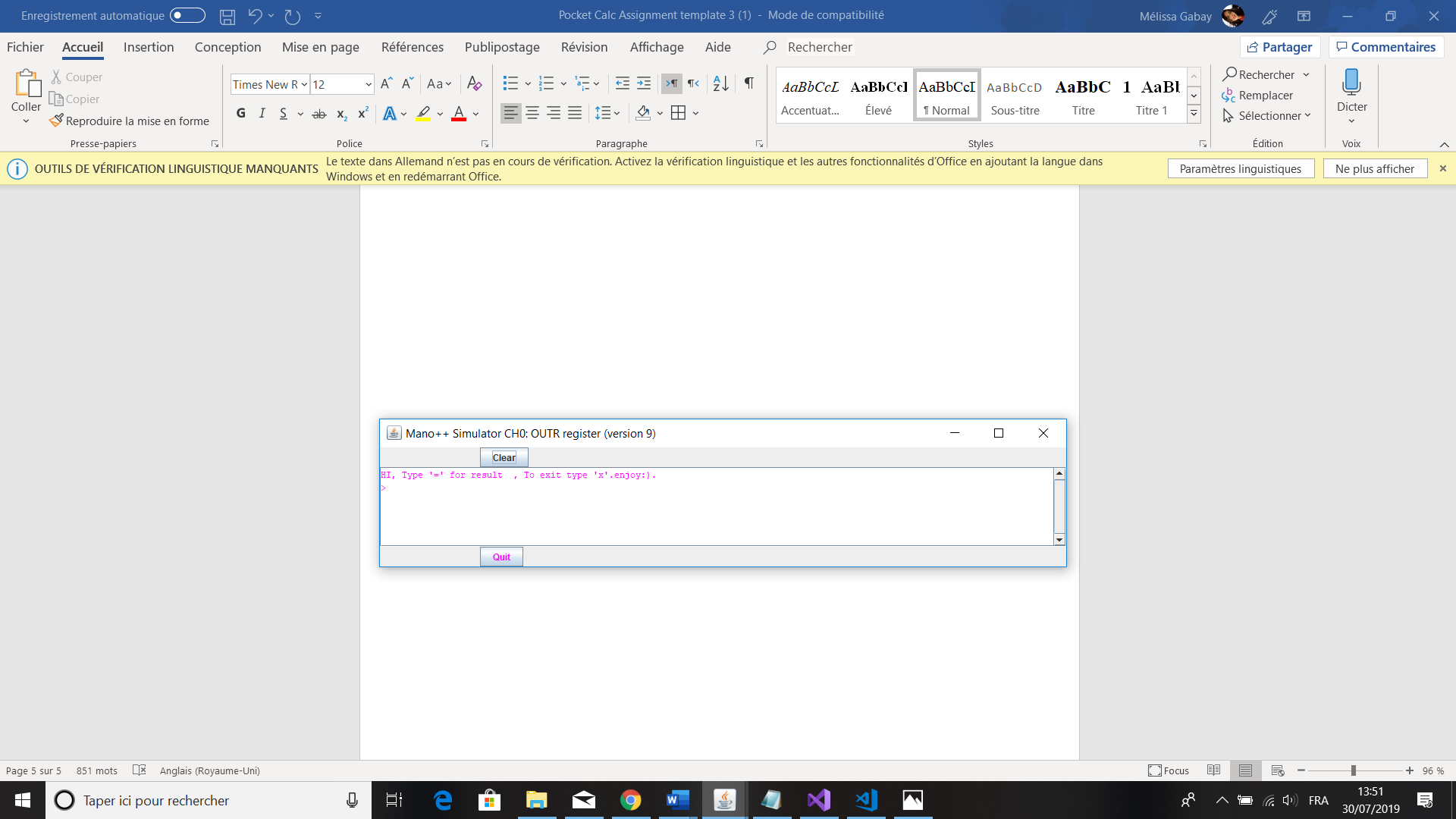
    }

}

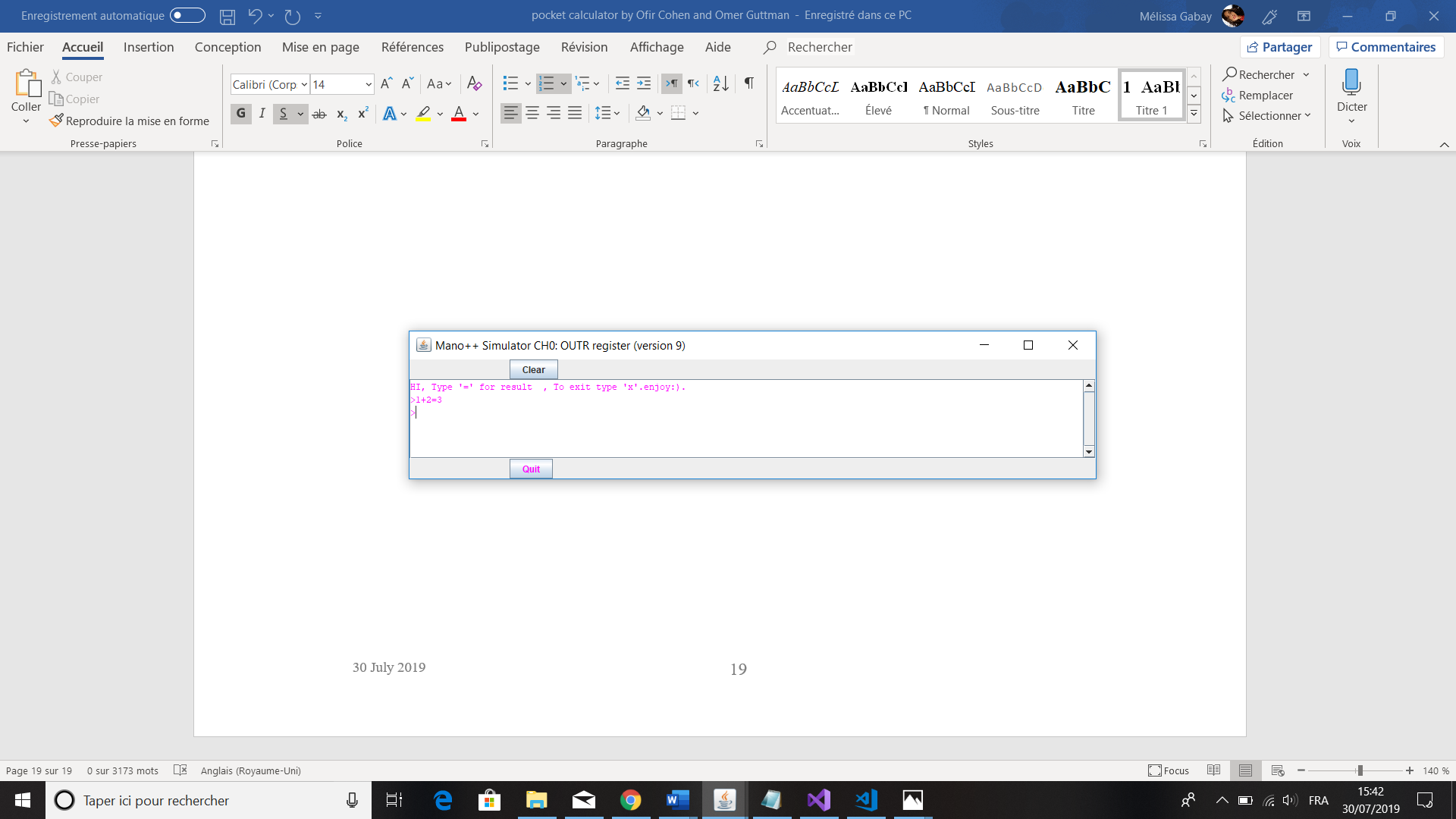
}

# The User Guide:

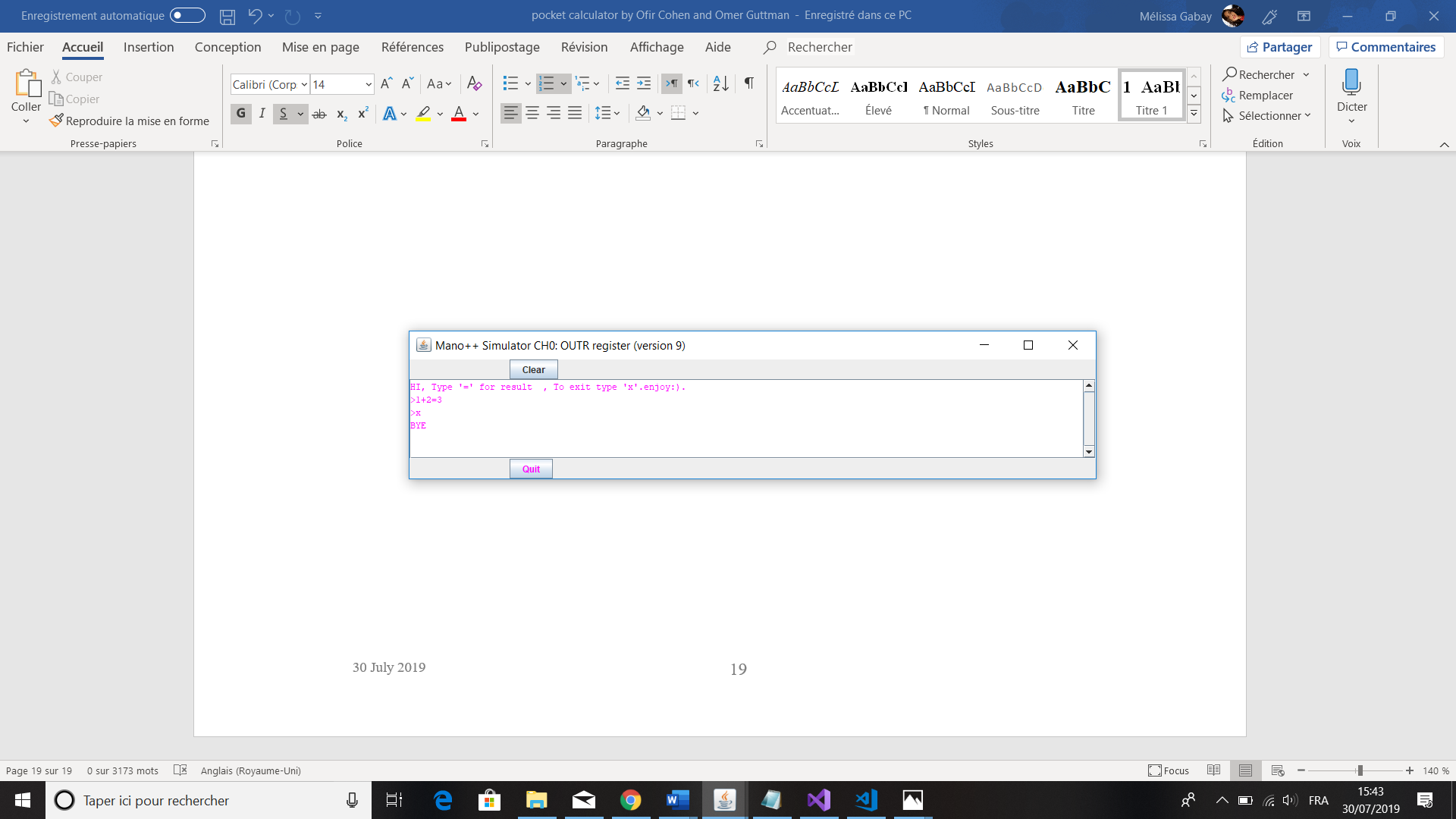
* + 1. The program welcomes the user asking him to write a correct arithmetical operation and to type "=" at the end of the operation.



* + 1. The user enters the operation of his choice (addition, subtraction, multiplication or division).



* + 1. To exit, enter the letter "x" and the program will send you "BYE" and the user can close the program.



# 

# The Mano Code

**\*Assembly code attached**