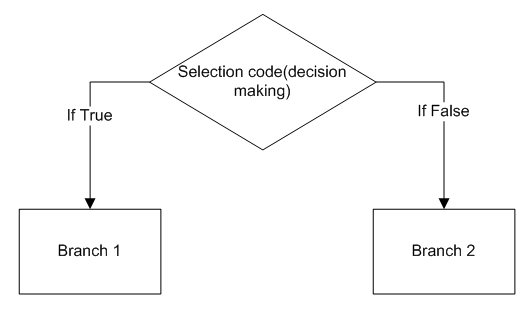
**M3 unit 19**

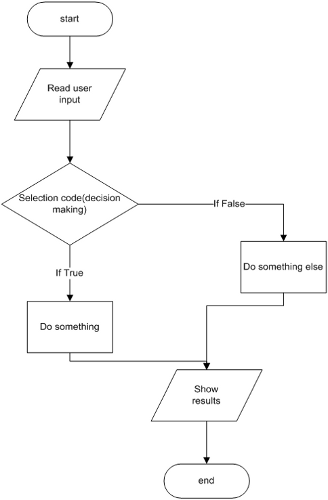
**Task 1**

**Branching**: one of the techniques used in assembly programming. Therefore, the purpose of branching in programming is to perform the program code in a sequence and the program flow can be altered by a conditional statement because it is necessary to change the flow of the actual flow of the instruction. For example, a simple branching program aim was to jump from one component of the instruction to another where the program accurately jumped to a particular memory address, if the outcome of the pervious instruction was zero.

Most computer programs use if statement. So the purpose of if statement is to allow a programmer to choose a course of action dependent on the conditions given in the program. As a result the programmer will use the if statements to control the flow of the actual program. For example the programmer might choice to use an if statement because if the programs condition links with the statement at that moment the code will be performed and executed if not the program will be overlooked and the rest of program will be executed.

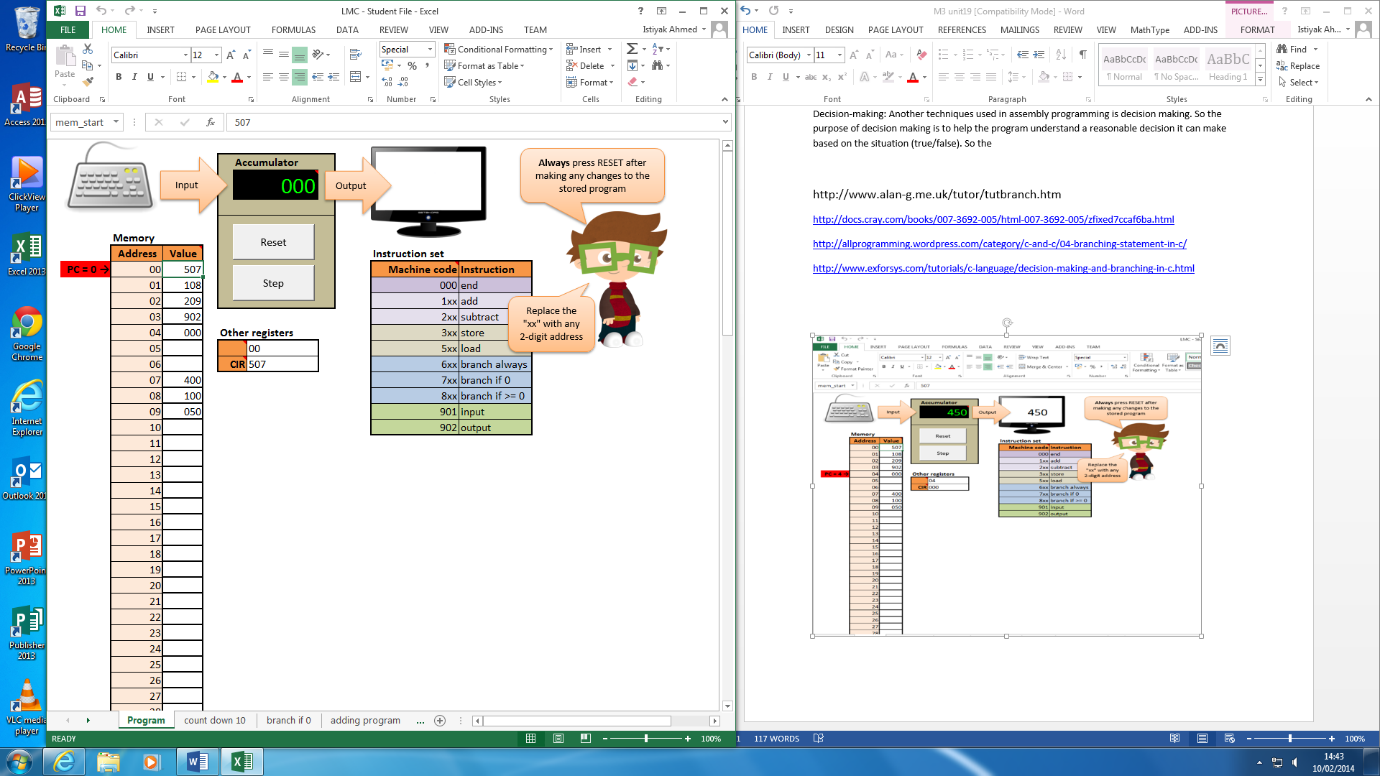
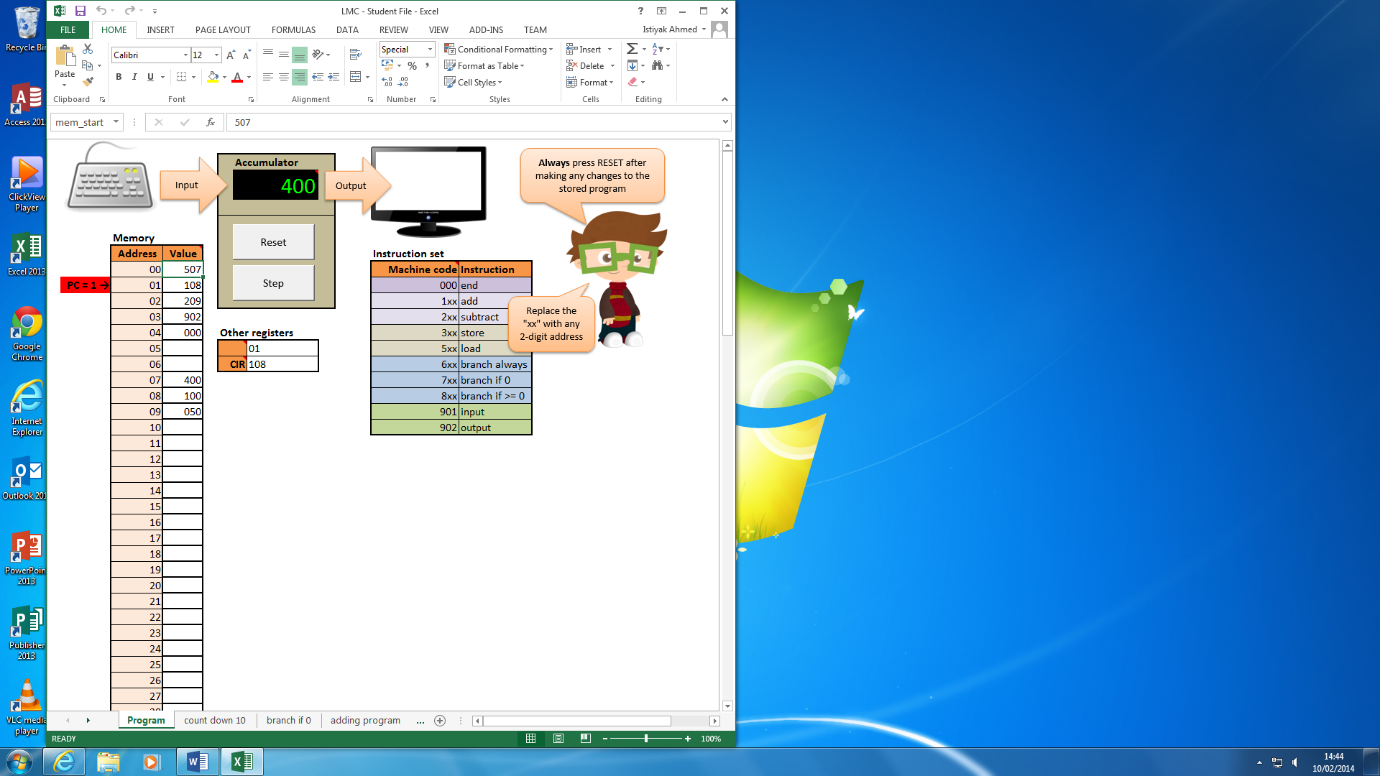
In this diagram it shows a connection between decision making and branch. So the objective of the program is to select the correct branch for execution, dependent on the outcome of the conditional statement.

**Decision-making**: Another techniques used in assembly programming is decision making. So the purpose of decision making is to help the program understand a reasonable decision it can make based on the situation (true/false). So for example if the programmer wants to provide a choice to the user to add or subtract a certain number then the actual program should permit the user to make a decision so the program can achieve the actual task in the program. This is why decision- making is an important process for every program relating to decision-making program because sometimes the users might encounter many problems in the program and the solution is to make a logical decision in order to resolve these precise problem.

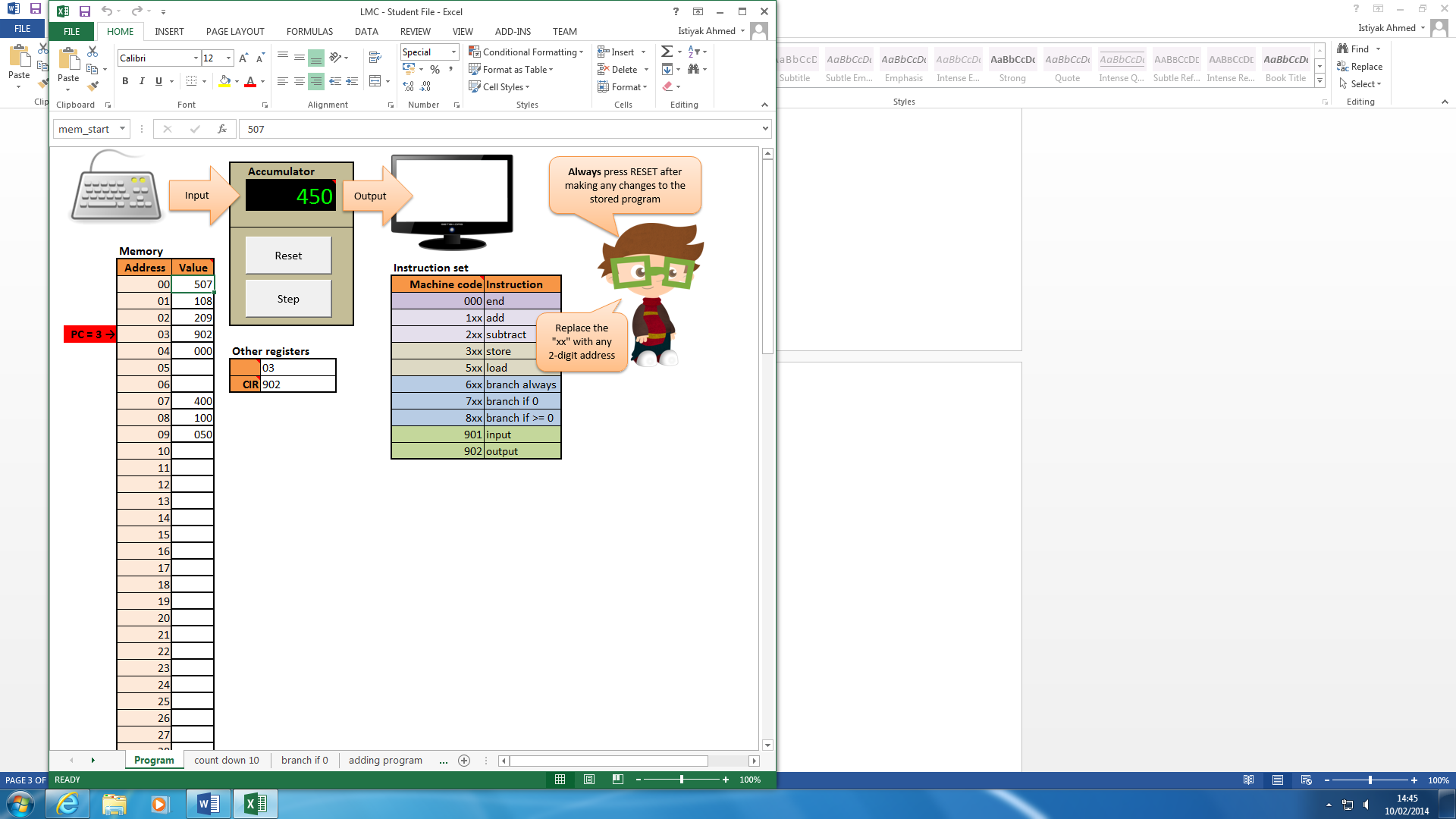
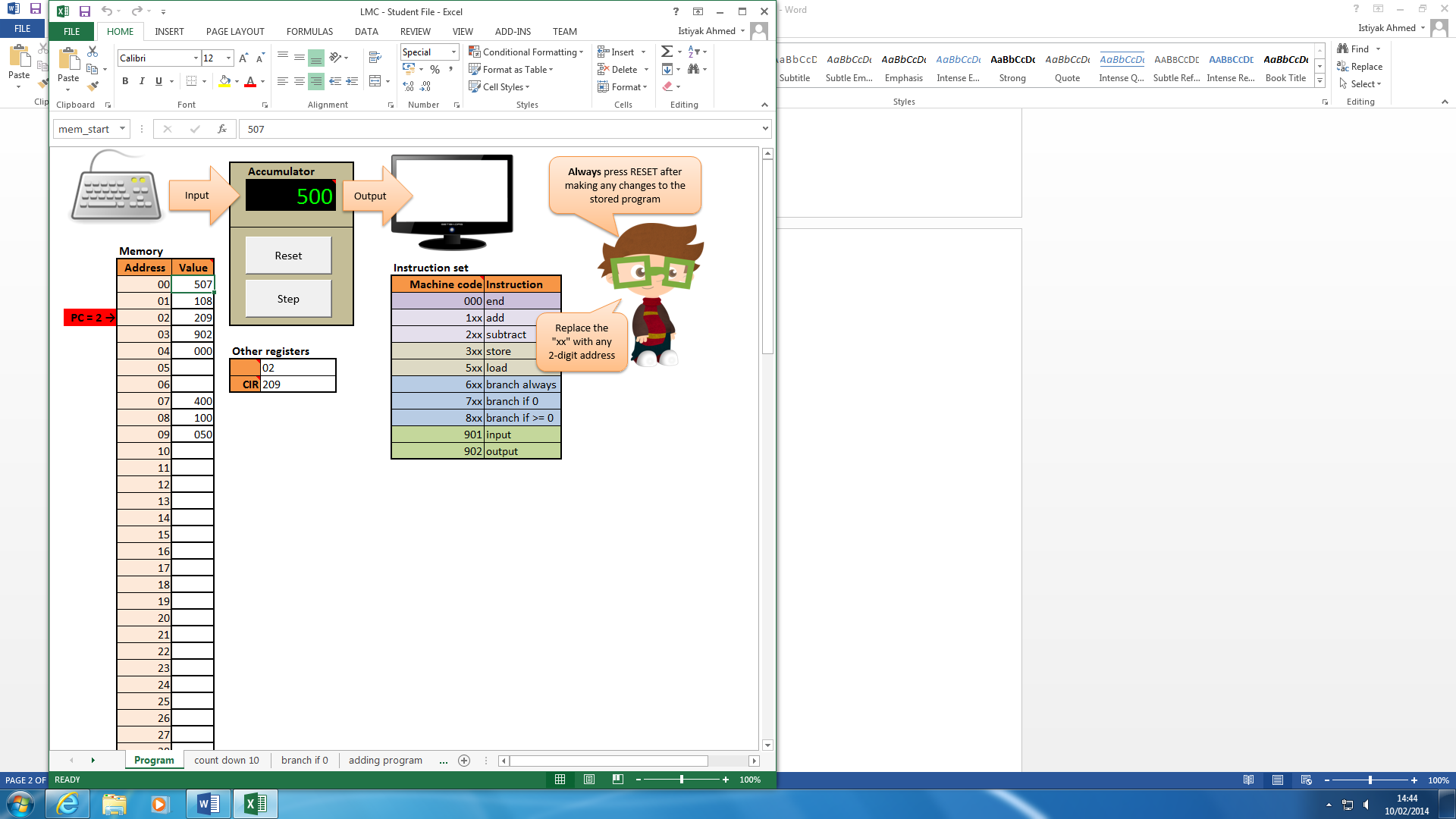


In this diagram it shows a decision –making process. So the links to conditional statements because it must not be ignored in order to function the actual program.

**3. Write a program that will add two numbers and then subtract a number and display the final result (demo & screen shots).**

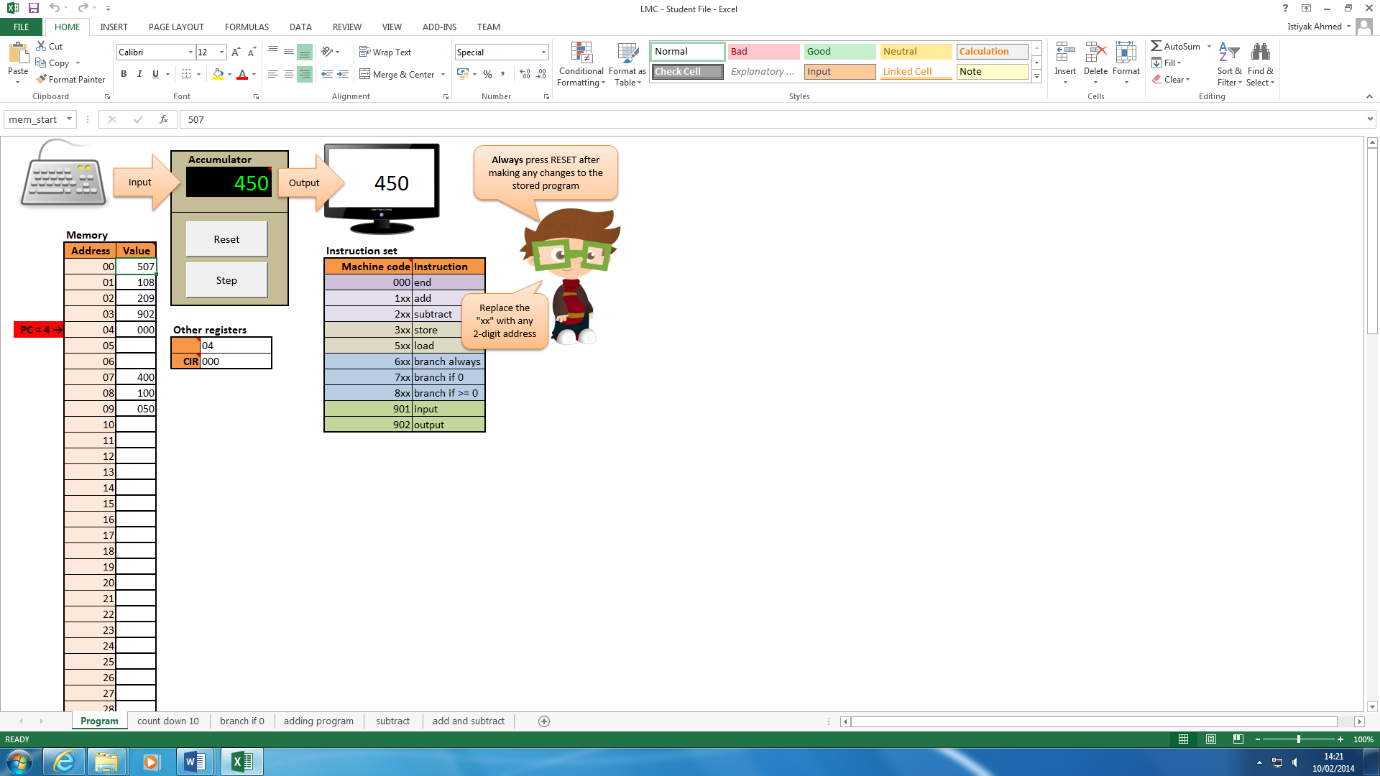
**Step 1** **Step 2**

In this screenshot this screenshot the program starts to load the content in address 7 which has a value of 400 which is hold in the accumulator

**Step 3** **Step 4**

In this screenshot this screenshot the program starts to add the content in address 08 which has a value of 100 to the accumulator. So it adds 400 +100=500. Then the program subtracts the content in address 09 which has value of 50 then the program subtracts 500-5=450.

**Step 5**



In this screenshot the result held in the accumulator is outputted to the screen.

**5. This flow chart shows the program that allows the user to enter a number, which will then count down to zero and then**

Step 1: start

Step 2: Input-user will input any integer value

Step 3: Output-display the value held in the accumulator.

Step 4: Process- subtract the content stored in address 18 from value in accumulator.

Step 5: Output-display the result held in the accumulator.

Step 6: Branch to address 17 if value is 0

Step 7: Branch always to address 2.

Step 8: End

Step 1 -Start

Step 2- Input

Step 3-Output

Step 4 -Process

Step 5 -Output

Step 6-Branch if value is 0

Step 8-End

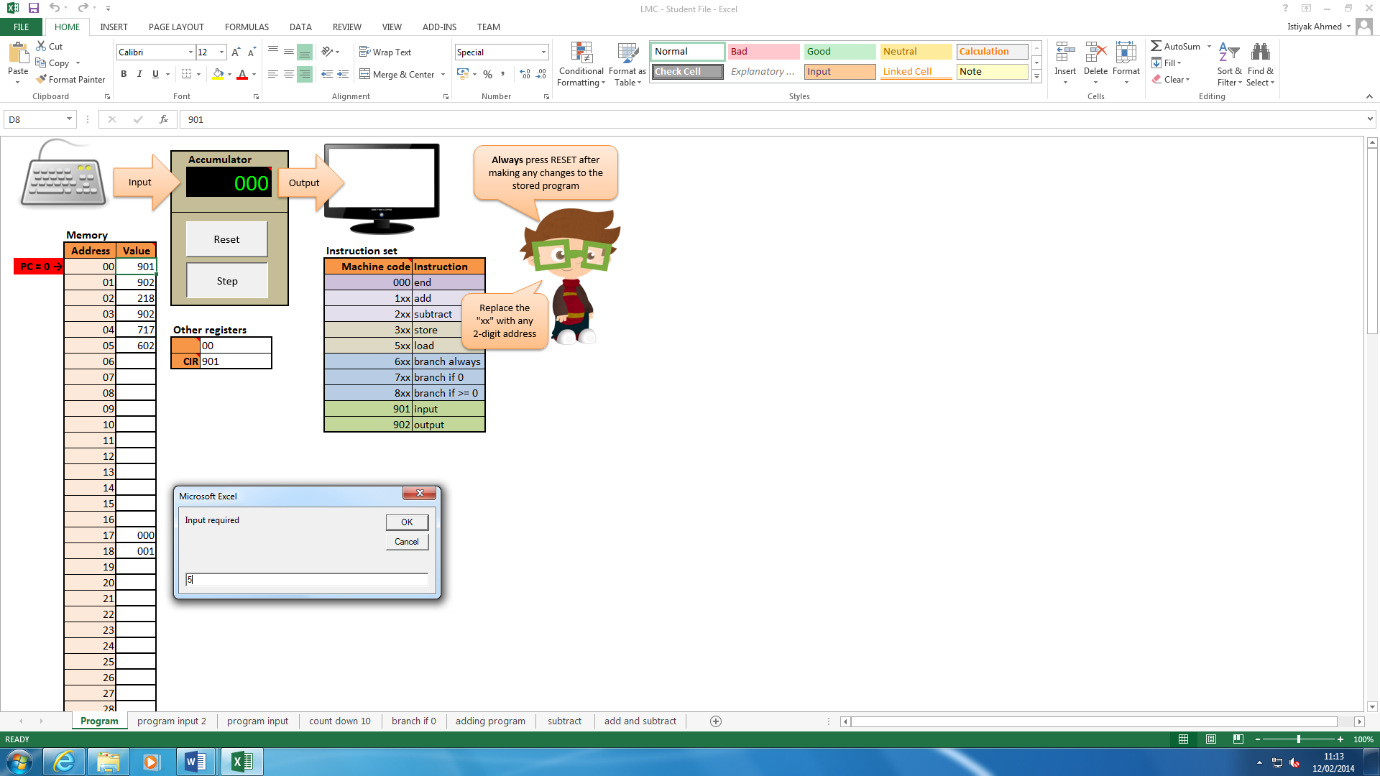
Value = 0

Value > 0

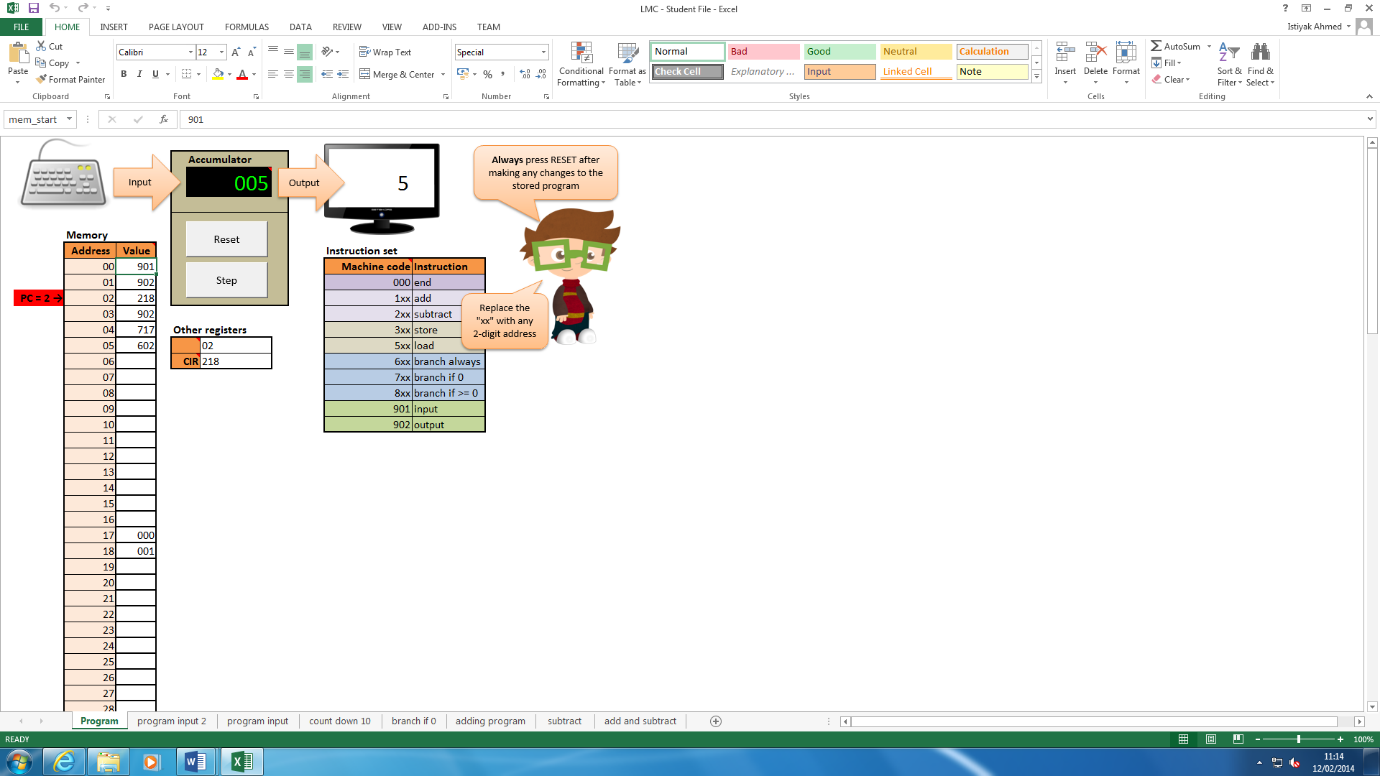
Step 7-Branch always

**A program that will allow the user to enter a number, which will then count down to zero**

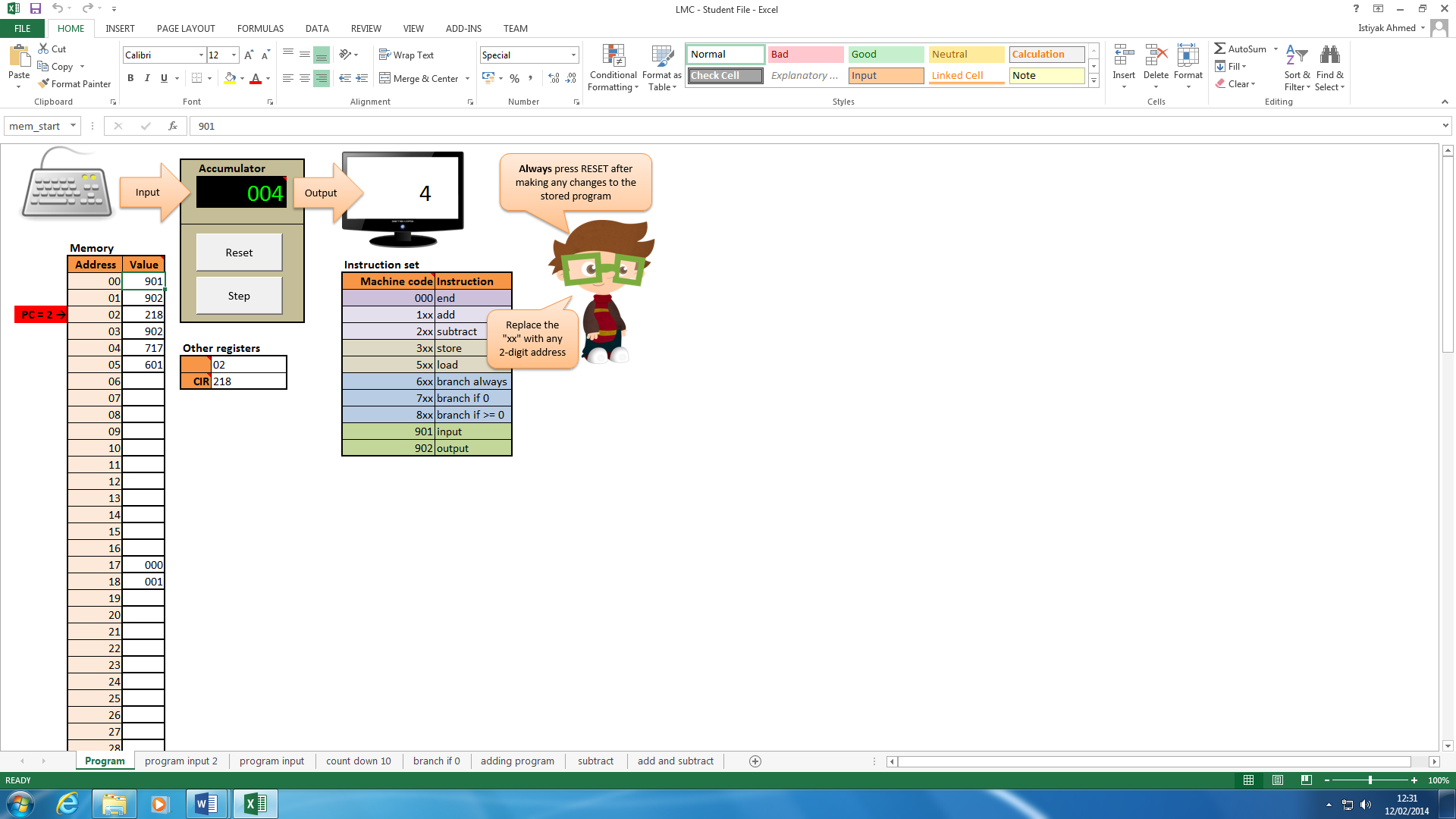
**Step 1**



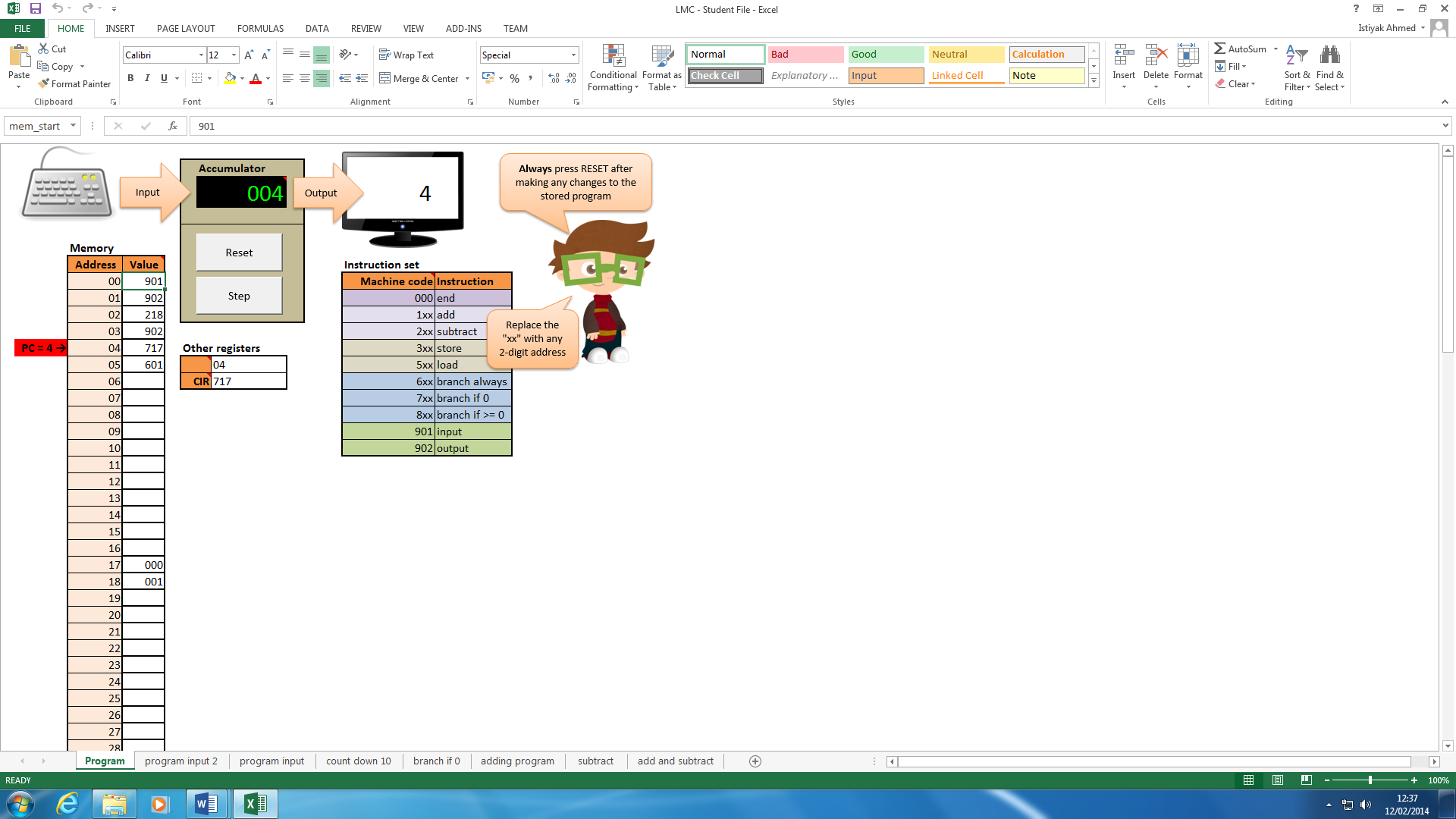
Step 1: in this screenshot the program needs to input any integer value the user wants, I have entered in 5, when the user clicks ok the program counter on will move down to address ‘01’ to perform the next step..

**Step 2**

Step 2: in this screenshot the program counter has moved down and the accumulator is now holding the value of 5 and outputting on the monitor screen.

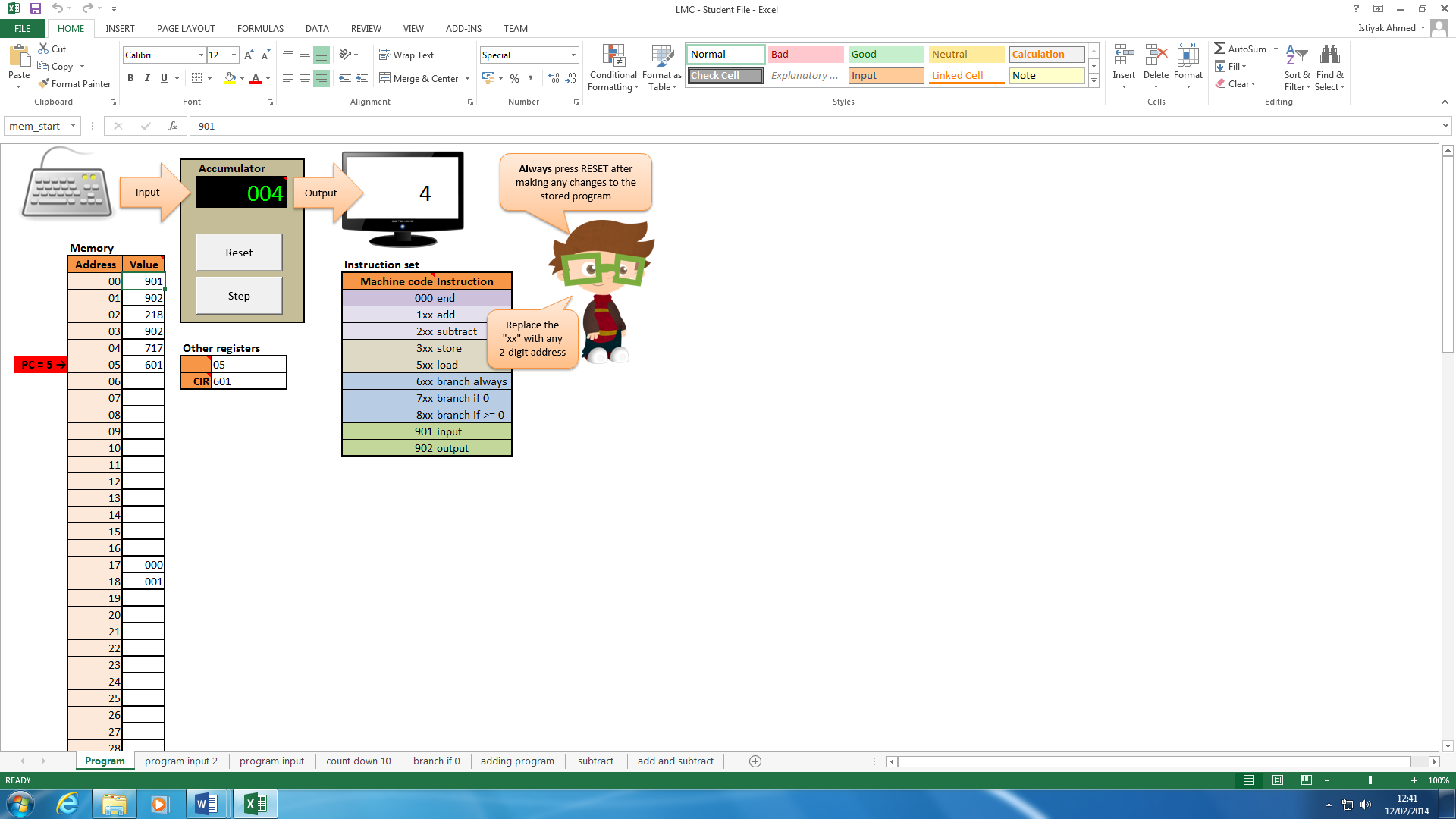
**Step 3**

Step 3: In this screenshot then subtracts away the value which is held within address ’18’ from the value which is held in the accumulator and the answer is 4. Finally it is outputted on the monitor screen which is held in the accumulator.

**Step 4**

Step 4: in this screenshot the program is executing the instruction which is held in address 04 and this instruction is branch to address 17 if the value is 0 but since the value is 10 the program will go to address 05 next. The value in the accumulator is 4 will be sent to address 05 next.

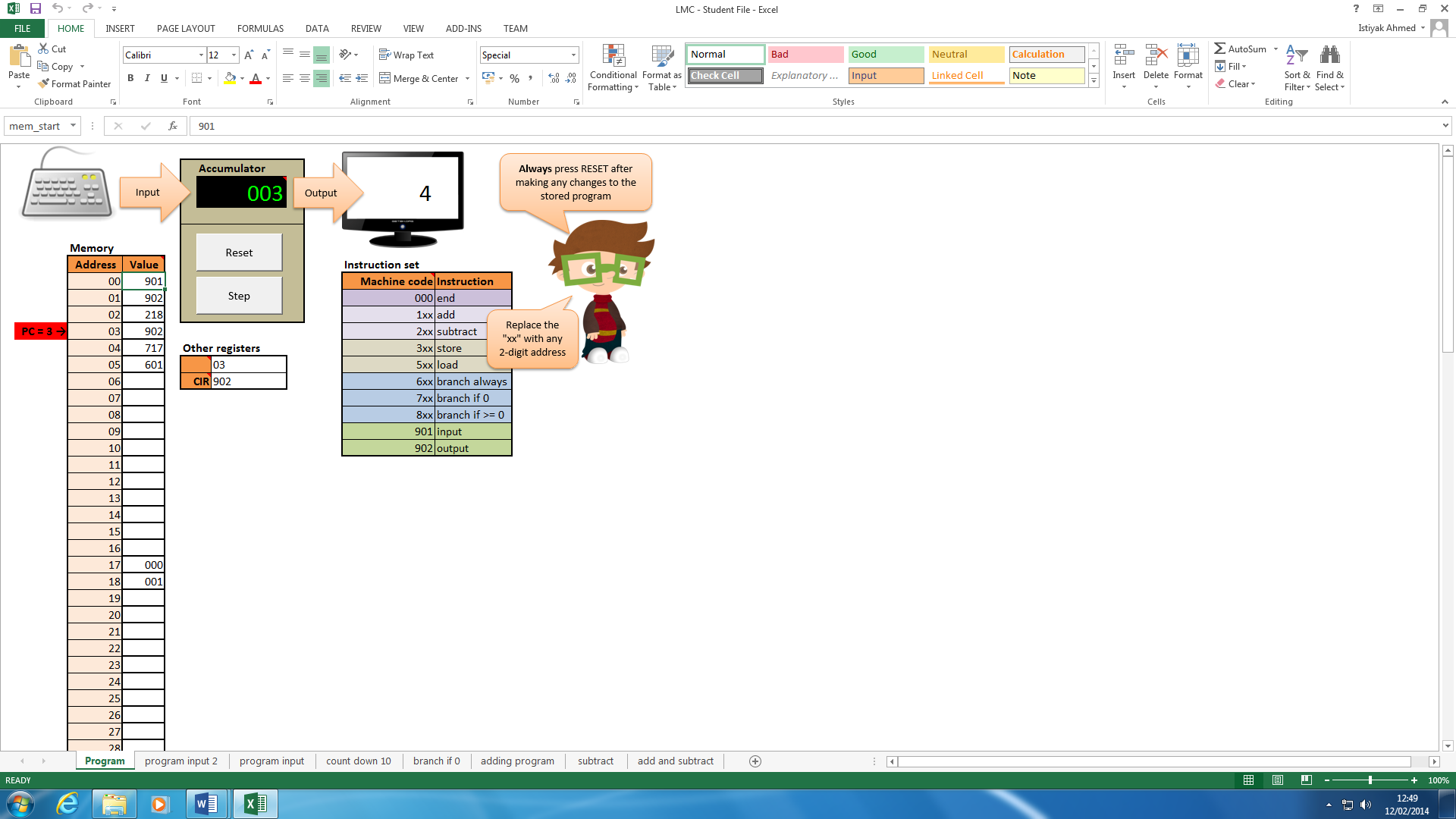
**Step 5**

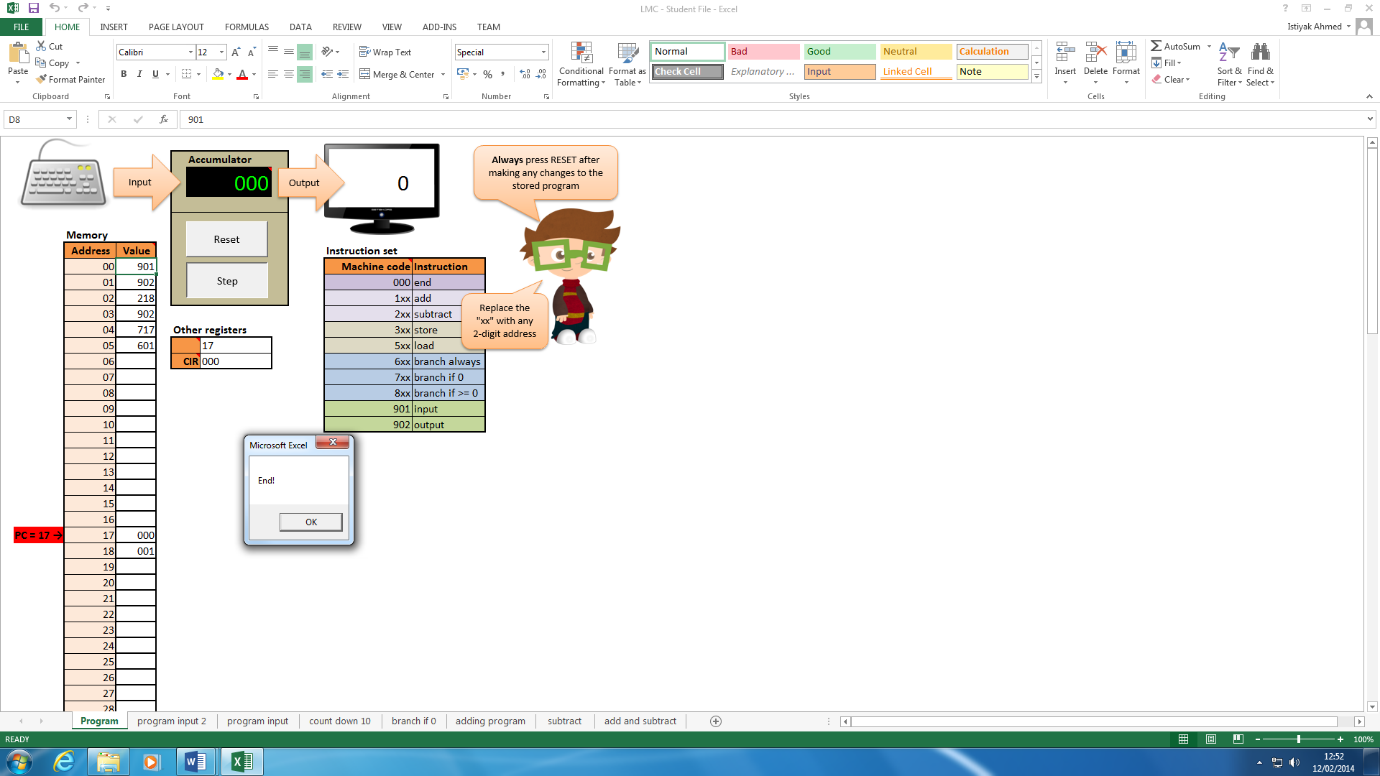


Step 5: In this screenshot the program is now on address 05 and the instruction held in this address is branch always to address 01.

**Step 6**

Step 6: In this screenshot the program loops back to address 02 also it keeps performing every instruction and now the previous steps I explained will wait until it reaches the value to go to 0.



**Step 7**

Step 8: In this screenshot the value is now on zero and is on address 04, the instruction held in this is address is branch to address 17 if value is 0 and since it is zero the program will jump.

Also in the screenshot, in this address the program ends smoothly.

**References**

http://www.alan-g.me.uk/tutor/tutbranch.htm

http://docs.cray.com/books/007-3692-005/html-007-3692-005/zfixed7ccaf6ba.html

http://allprogramming.wordpress.com/category/c-and-c/04-branching-statement-in-c/

http://www.exforsys.com/tutorials/c-language/decision-making-and-branching-in-c.html