**4CS015 Fundamentals of Computing – Workshop-6**

**Workshop tasks:**

1. Von Neumann Simulator. This program simulates a very simple computer with the von Neumann architecture.

a. Download the von Neumann Simulator (VonNeumann.exe) program from WOLF in the Week 5 folder. Save it in your Documents folder and run it. You will see a window like this:

Diagram

Description automatically generated

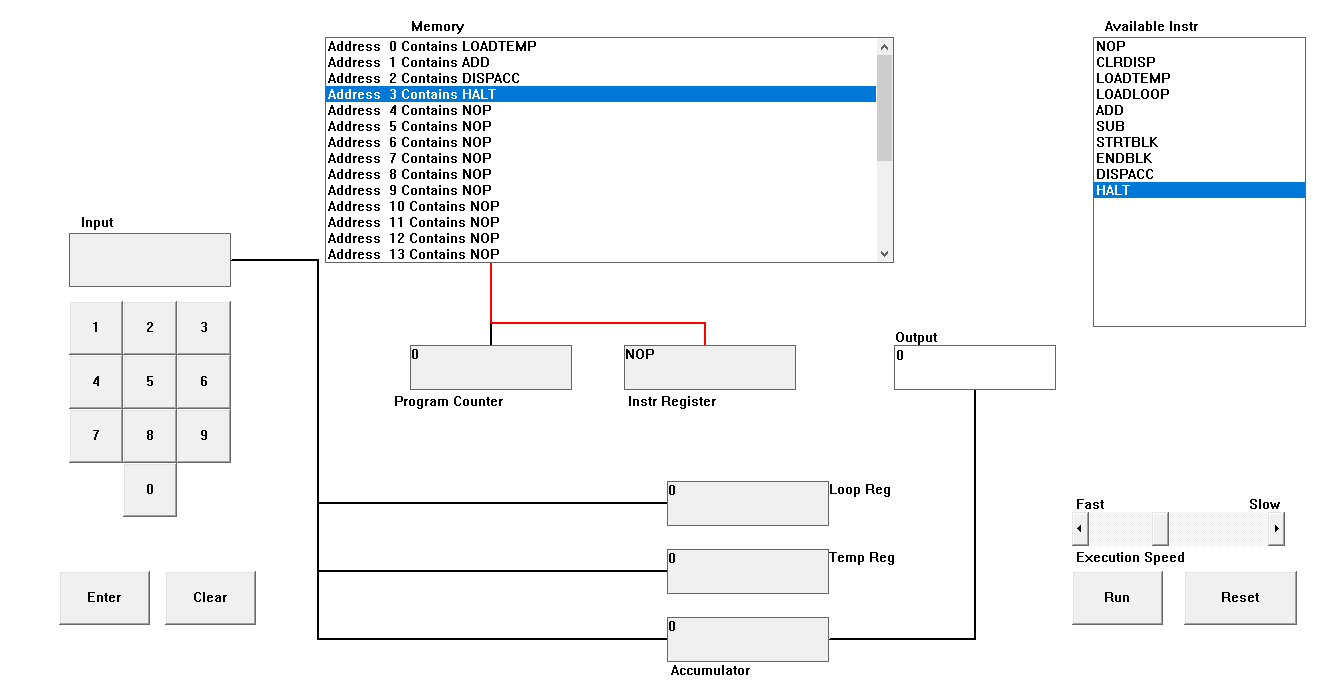
The simulator has a small program memory area which is available for programming. To enter your program instructions simply click on the “Available” instruction on the list on the right and then click on the “Memory” location you wish to put it in. 

This simulator understands only the following ten instructions:

|  |  |
| --- | --- |
| NOP | No Operation, i.e. do nothing. |
| LOADTEMP | Get a number from the keypad, completed by the Enter key, into the Temporary Register. |
| LOADLOOP | Get a number from the keypad, completed by the Enter key, into the Loop Register. |
| CLRDISP | Clear the Display. |
| ADD | Add the Temporary Register to the Accumulator |
| SUB | Subtract the Temporary Register from the Accumulator |
| DISPACC | Display the contents of the Accumulator |
| STRTBLK | Start of Loop Block |
| ENDBLK | End of Loop Block |
| HALT | Halt. Stop Program |

Load the following program into the memory:

LOADTEMP  
ADD  
DISPACC  
HALT



To do this, first click on the “LOADTEMP” in the list of instructions on the right of simulator window. Then click on Memory location with “Address 0 Contains NOP”. This will then change into “Address 0 Contains LOADTEMP”. Repeat the process with “Address 1” and so on until the whole program is loaded.

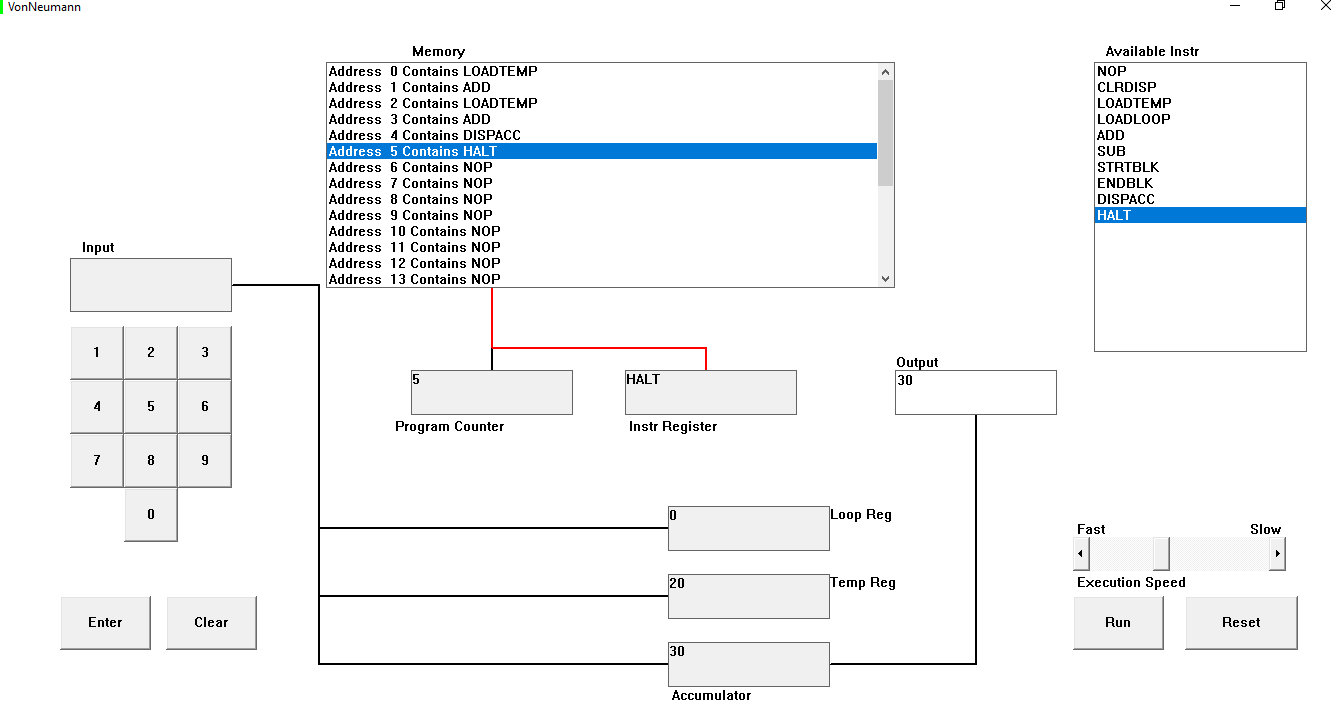
Run the program by clicking on the “Run” button. The simulator would highlight the Address 0 location and then pause. It is executing the instruction “LOADTEMP” which requires you to input a number into the keypad.   
  
Click 2 or 3 numbers on the keypad and then click the “Enter” button. The simulator will then resume running the program and execute the instruction “ADD”. This adds the number that you just entered, to the zero in the accumulator.   
  
The next instruction is “DISPACC” which stands for “Display Accumulator”, and it does exactly that. After than the simulator stops running the program when it executes the instruction “HALT”.

d. Load the following program into the simulator:

**Ans:**

LOADTEMP  
ADD  
LOADTEMP  
ADD  
DISPACC  
HALT

What do you think it does? Write your answer below (10 marks)



In the above program two numbers are added 10 and 20 in keypad. Instructions are LOADTEMP, ADD, LOADTEMP, ADD, DISPAC AND HALT is loaded to the memory respectively. Then, the input is taken as 10 from LOADTEMP in address 0 which is placed in Temp Reg (Temporary Register). The number is then added with 0 due to ADD instruction in address 1. Again, the LOADTEMP in address 3 is activated which prompts the user for the input. The input two is taken as 20. The ADD instruction in address 3 adds the input one and input two GIVING THE RESUSLT 30. The added value is then stored in accumulator. Finally, DISPACC sends the data to the output and the program ends after HALT instruction.

e. Write a program to add 3 numbers together. List your program below (10 marks)

**Ans:**

LOADTEMP

ADD

LOADTEMP

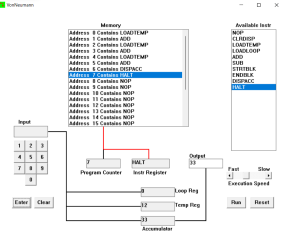
ADD

LOADTEMP

ADD

DISPACC

HALT



In the above Von Neumann Simulator, three numbers i.e., 10, 11, and 12 are added. This is done using LOADTEMP, ADD, DISPACC and HALT instructions in the memory. Firstly, we use LOADTEMP in address 0 to input our first number i.e., 10 in the temporary register. The number is then added with 0 due to ADD instruction in Address 1. This process is again repeated, and we input 11 in the LOADTEMP in Address 2, which is then added with 10 using ADD instruction in Address 3. This operation results with 21 in temporary register. We again repeat these instructions with LOADTEMP in Address 4 and ADD in Address 5 which adds our final input 12 with the output 21. Finally, the output 33 is shown using DISPACC in Address 6 and the program ends after HALT instruction in Address 6.

f. Write a program to subtract a number from another. List your program below (10 marks)

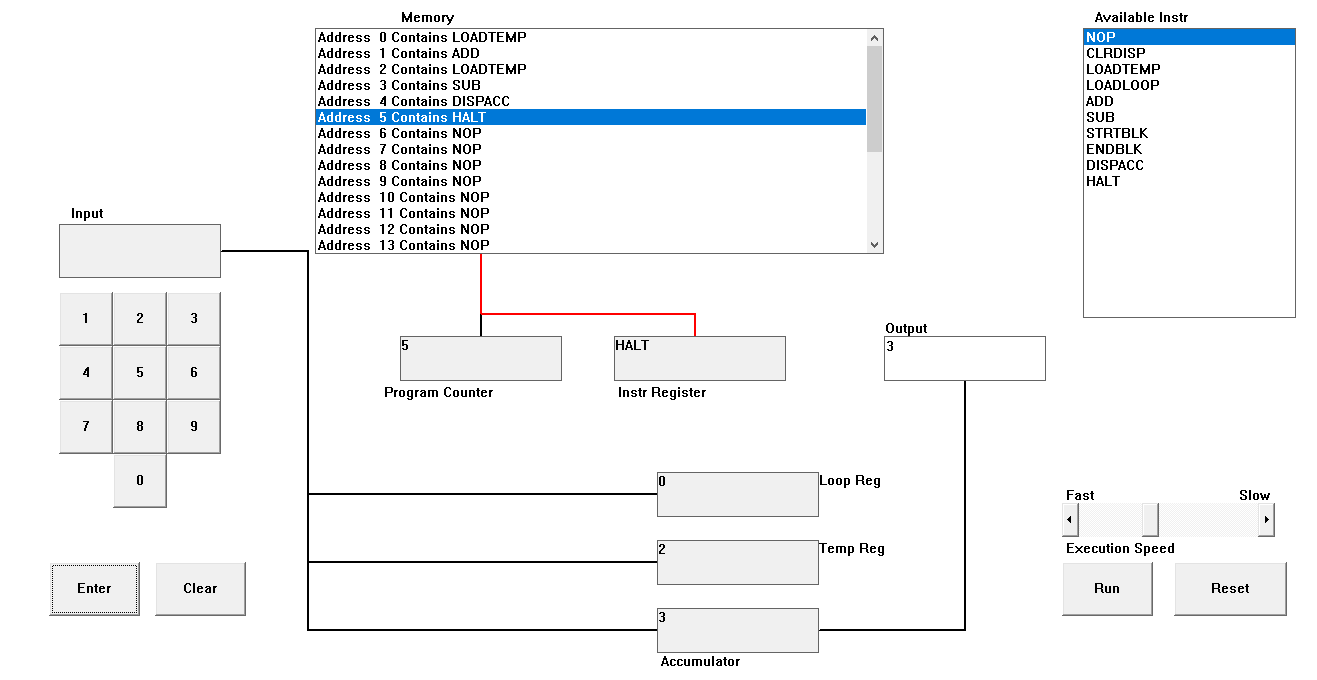
**ANS:**

LOADTEMP

ADD

LOADTEMP

DISPACC

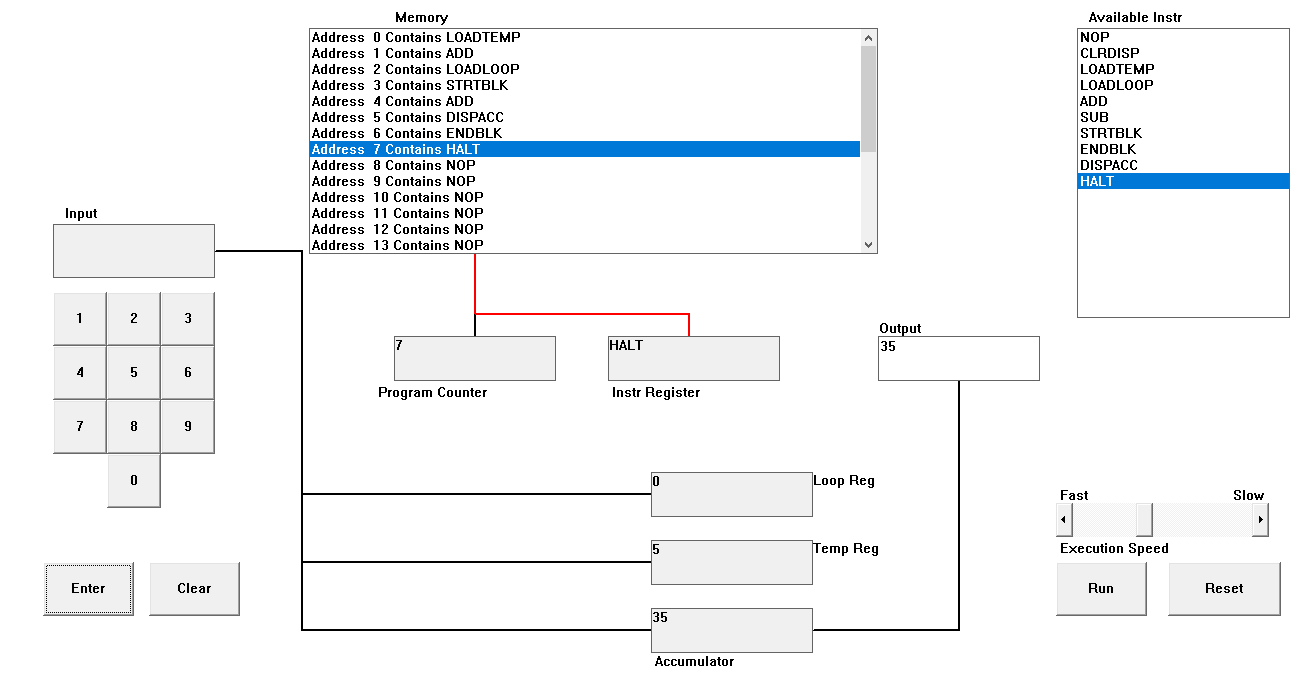
HALT  
  


In the given program we have subtract the numbers. I use LOADTEMP, ADD, SUB, DISPACC and HALT. First, I just put the instruction on the memory location then I Run the program putting number on KEYPAD. On the memory location first, the program take add then it subtracts that number, and it gives the output. In this program I enter 5 and 2 at last it’s given me the output 3 because first input takes add then second input subtract with first number then the program stops by the help of Address 5 HALT and store the data on the accumulator.

g. Load the following program into the simulator:

ANS:

LOADTEMP  
ADD  
LOADLOOP  
STRTBLK  
ADD  
DISPACC  
ENDBLK  
HALT



Run it and when it reaches the LOADTEMP instruction, enter 5 on the keypad and click the “Enter” button. When it reaches the LOADLOOP instruction, enter 6. What do you think the program does? Write your answer below in the form of an equation (10 marks)

**EQN:**

5 + ( 5 \* 6 )

In this program the instruction is LOOPTEMP, ADD, LOADLOOP, STRTBLK, ADD, DISPACC, ENDBLK, HALT. These instructions are used in Memory location which the program runs. STRTBLK start the loop block and ENDBLK stop the loop block. First of all, we put the adding num on address 0 LOOPTEMP then the program adds the number and it ask the number to loop on address 3 LOOPTEMP. After that address 3 STRTBLK which start loop block given by the user. After that it ADD and DISPACC then address 6 ENDBLK stop the loop block and HALT stop the program.

h. Write a program that will let you add 5, or 10 or 20 numbers together. List your program below and explain how it works (25 marks)

Adding 5 number together

**ANS:**

LOADLOOP

STRTBLK

LOADTEMP

ADD

DISPACC

HALT  
Diagram

Description automatically generated

In the given program instruction is LOADLOOP, STRTBLK, LOADTEMP, ADD, DISPACC, HALT. The input number is 2 which the number is added 5 times in this program. First, we put the instruction to memory address then we run this program. In the first address 0 LOOPLOOP we input the number 5 because in this address the number is repeated given by the user. Then it makes a loop block from address 1 STRTBLK after that we input the number 2 on address 2 LOADTEMP after that the given input 2 will be loop 5 times then it shifts to DISPACC after that address 5 ENDBLK stop the loop then HLAT stop the program. Then its store output in the Accumulator which will be 10.

Diagram

Description automatically generated

In the given program instruction is LOADLOOP, STRTBLK, LOADTEMP, ADD, DISPACC, HALT. The input number is 2 which the number is added 10 times in this program. First, we put the instruction to memory address then we run this program. In the first address 0 LOOPLOOP we input the number 5 because in this address the number is repeated given by the user. Then it makes a loop block from address 1 STRTBLK after that we input the number 2 on address 2 LOADTEMP after that the given input 2 will be loop 5 times then it shifts to DISPACC after that address 5 ENDBLK stop the loop then HLAT stop the program. Then its store output in the Accumulator which will be 20.

Diagram

Description automatically generated with medium confidence

In the previous program same instruction are used. The input number is 2 which the number is added 20 times in this program. First, we put the instruction to memory address then we run this program. In the first address 0 LOOPLOOP we input the number 5 because in this address the number is repeated given by the user. Then it makes a loop block from address 1 STRTBLK after that we input the number 2 on address 2 LOADTEMP after that the given input 2 will be loop 5 times then it shifts to DISPACC after that address 5 ENDBLK stop the loop then HLAT stop the program. Then its store output in the Accumulator which will be 40.

i. Write a program that will let you multiply 2 numbers together. List your program below and explain how it works (35 marks)

**ANS:**

LOADLOOP

STRTBLK

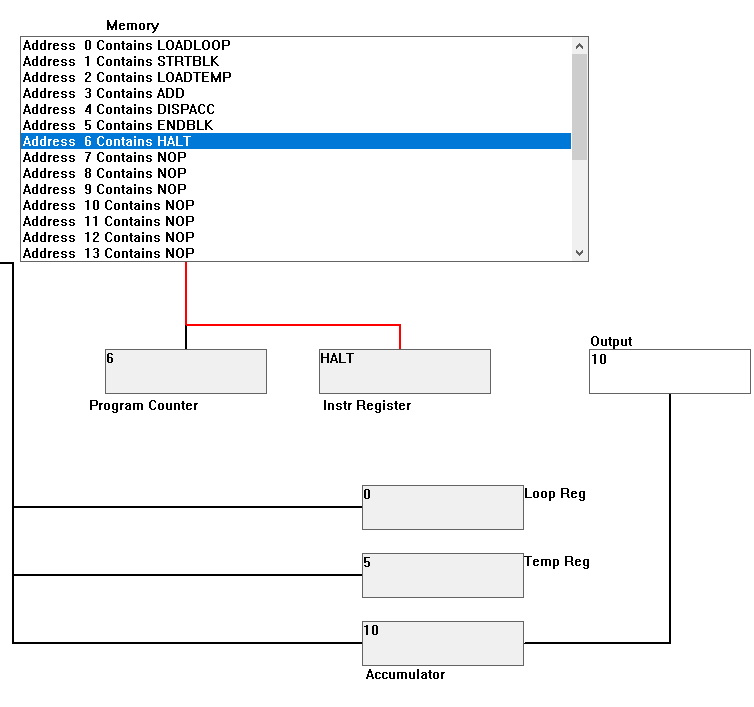
LOADTEMP

ADD

DISPACC

ENDBLK

HALT



**EQN:**   
  
5 \* 2 = 10

In this program we have to multiply 2 number together. As, we can see in the OUTPUT answer is 10.

The Instruction are LOADLOOP, STRATBLK, LOADTEMP, ADD, DISPACC, ENDBLK, HALT. On the right side at the bottom there will be RUN and RESTART button for the program. Starting by putting the input value on the KEYPAD. As, we can see inserting 5 \* 2 by using the instruction. First of all, on address 0 LOADLOOP the input number 2 its program which is to get a number from the keypad into the loop register. It repeats the given number from the keypad. Then given input is 2 which mean double looping on the process. After that it shift on address 1 STRATBLK which it makes a lopping block. Then it shifted on LOOPTEMP which it. Get a number from the keypad and complete the enter key into the temporary register. In this program we input number 5. Then it shifts to address 3 ADD then its shift to address 4 DISPACC which it displays the constant of the accumulator. The program repeats 2 times because the process is in looping block. After that it shift to address 5 ENDBLK which it stops the looping block then it shifts to address 6 HALT which the program stops. At last, we can see the output will be 10 which it stored in accumulator.