Program Loops (Looping)

A looping or Program loop is a sequence of instructions that are executed many times, each time with a different set of data.

e.g:- A Program to add numbers in an array.

Array

2050 → 5 (Size of array)
2051 → 22
2052 → 03
2053 → 04
2054 → 01
2055 → 02

Output

3050 → Sum
3051 → Carry

Algorithm:—

1. Load the base address of the array in HL register Pair.
2. Use the Size of the array as a Counter
3. Initialise accumulator to 0.
4. Add content of accumulator with content stored at memory location given in HL Pair.
5. Decrease Counter on each addition.

Program:-

Address	mnemonics	operands/data	Comments
2000,01,02	LDA	2050	load A with content of 2050
2003	MOV	B, A	copy A to B
2004,05,06	LXI H	H, 2051	point out new Address 2051
2007,08	MVI	A 00	clear A
2009,0A	MVI	C, 00	clear C
200B	ADD	M	A → A+M (2051)
200C	INX	H	Increment in HL Pair
200D,0E,0F	JNC	2011	Jump if not Carry
2010	INR	C	Increment in C
2011	DCR	B	Decrement in B
2012,13,14	JNZ	200B	Jump if not Zero
2015,16,17	STA	3050	store A to 3050
2018	MOV	A, C	copy A ← C

2014	STA	3051	Store Content of A to 3051
	HLT		Terminate the Program

Explanation:

1. LDA 2050 :- load accumulator with Content of location 2050
2. MOV B,A :- Copy Content of accumulator to register B.
3. LXI H,2051 :- Store 20 to H register and 51 to L register to initialize memory 2051
4. MVI A,00 :→ Store 00 to accumulator.
5. MVI C,00 :→ Store 00 to register C
6. ADD M :→ Add accumulator with the Contents of memory location given in HL register Pair.
7. INX H :→ increment in HL Pair.
8. JNC 2011 :→ Jump to location 2011, if not carry, otherwise to the next location
9. INC C :→ increase Content of register C by 1.
10. DCR B :→ decrease Content of register B by 1.
11. JNZ 2008 :→ Jump to location 2008, otherwise to the next location.
12. STA 3050 :→ Store Content of accumulator to 3050 memory
13. MOV A,C :→ Copy Contents of register C to Accumulator.
14. STA 3051 :→ store Content of accumulator to memory 3051
15. HLT :→ Terminates the Program.

Programming Arithmetic and logic operations :→

A program to multiply two 8-bit numbers.

e.g Input :-	Output :-
2051 → D_1 (07)	3051 → Carry (01)
2050 → D_2 (43)	3050 → Result (25)

Algorithm :-

1. We are talking adding the number 43 Seven (7) times in this Example.
2. As the multiplication of two 8 bit numbers can be maximum of 16-bits so we need register pair to store the result.

Program :-

Address	mnemonics	operand/data	Comment
2000, 01, 02	LHLD	2050	load Content of 2050 to H load Content of 2051 to L
2003	XCHG		exchange HL with DE
2004	MOV	C, D	Copy D → C
2005, 06	MVI	D, 00	clear D
2007	LXI	H, 0000	clear HL Pair
200A	DAD	D	Add HL with DE, result in HL Pair
200B	DCR	C	Decrement in C
200C, 0D, 0E	JNZ	200A	Jump to 200A If not Zero
200F, 10, 11	SHLD	3050	Store Content of H to 3050 Store Content of L to 3051
2012	HLT		Terminate the Program.

Explanation :→

1. LHLD 2050 :- load Content of 2051 in H and Content of 2050 in L
2. XCHG :- exchange Content of H with D and Contents of L with E.
3. MOV C, D :→ Copies Content of D in C.
4. MVI D 00 :→ Assign Zero to D.
5. LXI H 0000 :→ Assign Zero to HL Pair.
6. DAD D :→ Add HL and DE and assigns the result to HL
7. DCR C :→ decrements C by 1.
8. JNZ 200A :→ Jumps Program Counter to 200A If Zero flag=0
9. SHLD :→ Store value of H at memory 3051 and L at 3050
10. HLT :→ Terminate the Program.

Note :- Program for the other arithmetic operations like add, subtract, multiply and divide by using diffe respective instruction can be developed in a similar fashion.

Logic operations :→ The basic computer has three machine instructions that perform logic operations : AND, CMA and CLA. All 16 logic operations can be implemented by software means because any logic function can be implemented by AND and Complement operations.

e.g :→ Input

3050 → D

Output

3051 → 1's complement

3052 → 2's Complement

Algorithm :-

1. Load the data from memory 3050 into Accumulator.
2. Complement Content of accumulator.
3. Store Content of accumulator in memory 3051 (1's)
4. Add 01 to Accumulator Content.
5. Store Content of accumulator in memory 3052 (2's)
6. Stop.

Program :- To find 1's and 2's Complement of 8-bit no.

memory	mnemonics	operands/data	Comment
	LDA	3050	load A with Content of 3050
	CMA		Complement of A
	STA	3051	Store result in 3051
	ADI	01	A → A+01
	STA	3052	Store result in 3052
	HLT		Terminate.

Explanation :→

1. A is an 8-bit accumulator which is used to load and store the data directly.
2. LDA is used to load accumulator direct using 16-bit address.
3. CMA is used to Complement Content of accumulator.
4. STA is used to store accumulator direct using 16-bit address
5. ADI is used to add data into accumulator immediately.
6. HLT Halt the Program.

*. The other logic operations can be implemented by using different Instructions in a similar fashion.