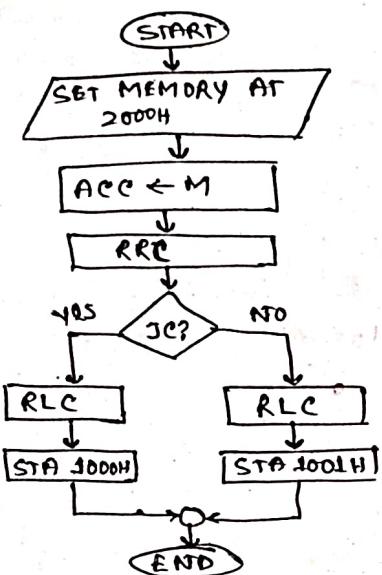


# Microprocessor Practical Question Answers

① Write an assembly language program for detection of even and odd data of a memory location.

⇒ **Flowchart:**



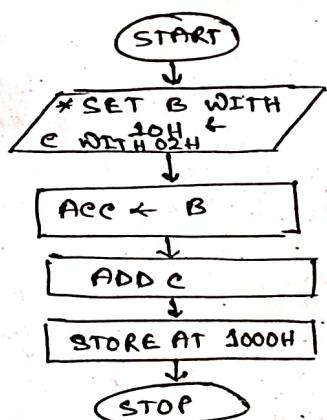
⇒ **Instruction:**

```

LXI H, 2000H
MOV A, M
RLC
JC LOOP
RLC
STA 1001H
LOOP: RLC
STA 1000H
HLT
  
```

② Write an assembly language program to add two 8-bit numbers.

⇒ **Flowchart:**



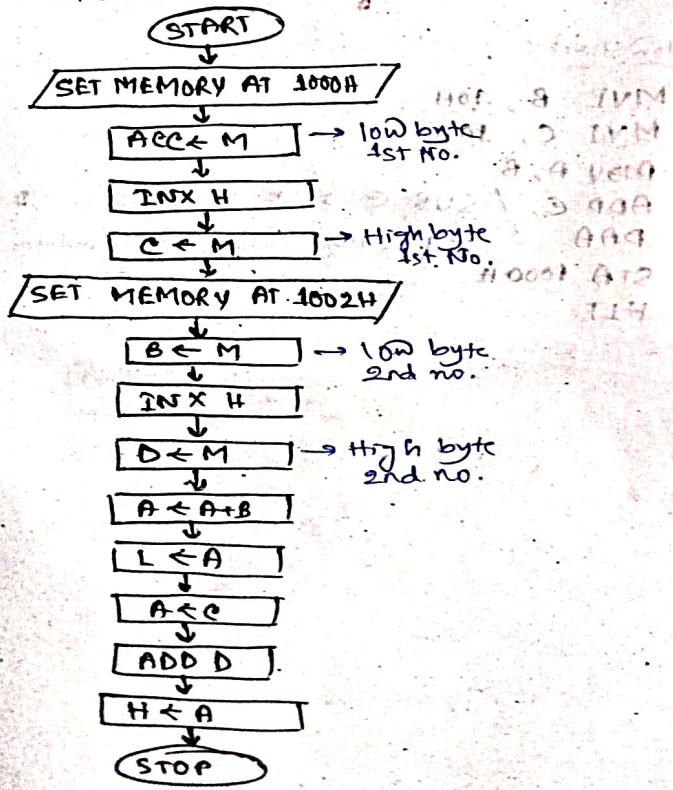
⇒ **Instruction:**

```

MVI B, 0AH
MVI C, 02H
MOV A, B
ADD C
STA 1000H
HLT
  
```

③ Write an assembly language program to add two 16-bit numbers.

⇒ **Flowchart:**



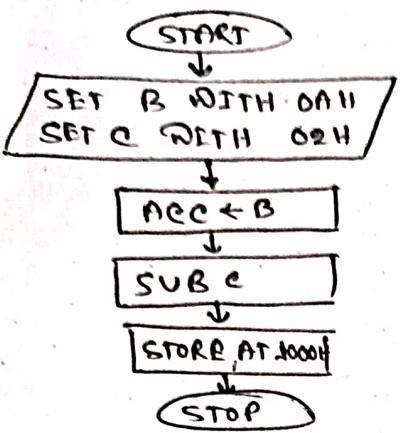
⇒ **Instruction:**

```

LXI H, 1000H
MOV A, M
INX H
MOV C, M
LXI H, 1002H
MOV B, M
INX H
MOV D, M
ADC B
MOV L, A
MOV A, C
ADC D
MOV H, A
HLT
  
```

④ Write an assembly language program to subtract two 8-bit numbers.

⇒ Flowchart:

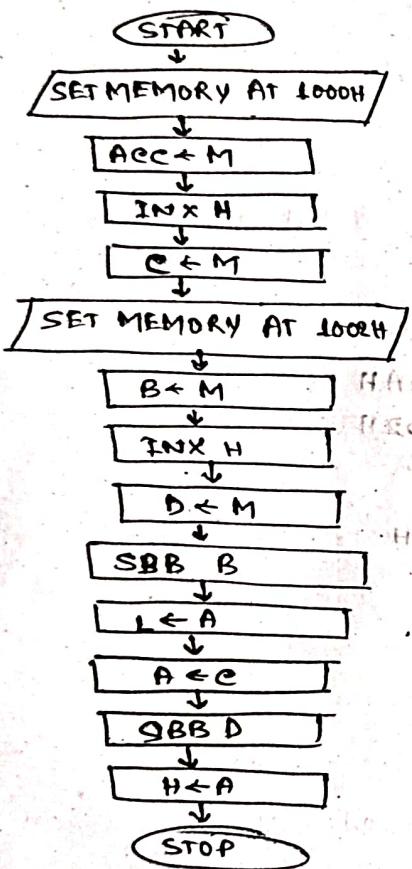


⇒ Instruction:

```
MVI B, 0AH  
MVI C, 02H  
MOV A,B  
SUB C  
STA 1000H  
HLT.
```

⑤ Write an assembly language program to subtract two 16-bit numbers.

⇒ Flowchart:

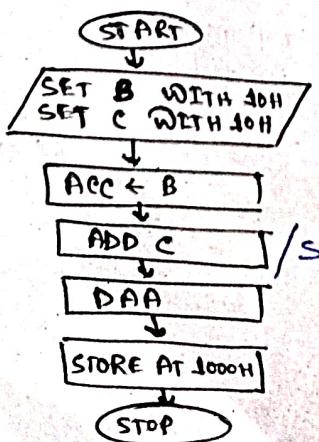


⇒ Instruction:

```
LXI H, 1000H  
MOV A, M  
INX H  
MOV M, M  
LXI H, 1002H  
MOV B, M  
INX H  
MOV D, M  
SBB B  
MOV L, A  
MOV A, C  
SBB D  
MOV H, A  
HLT.
```

⑥ Write a program to add two BCD Numbers.

⇒ Flowchart:



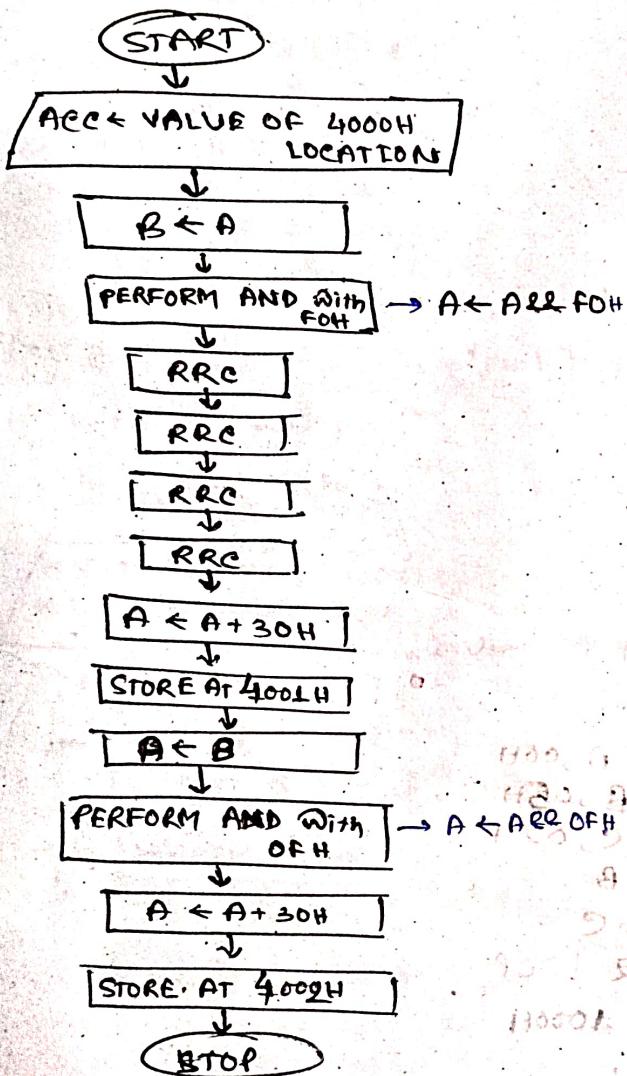
⇒ Flowchart:

```
MVI B, 10H  
MVI C, 10H  
MOV A,B  
ADD C / SUB C ; for subtraction  
DAA ; Decimal Adjust Accumulator  
STA 1000H  
HLT.
```

(HLT.)

⑧ Write a program to convert a BCD number into ASCII equivalent.

⇒ flowchart:



⇒ Instruction:

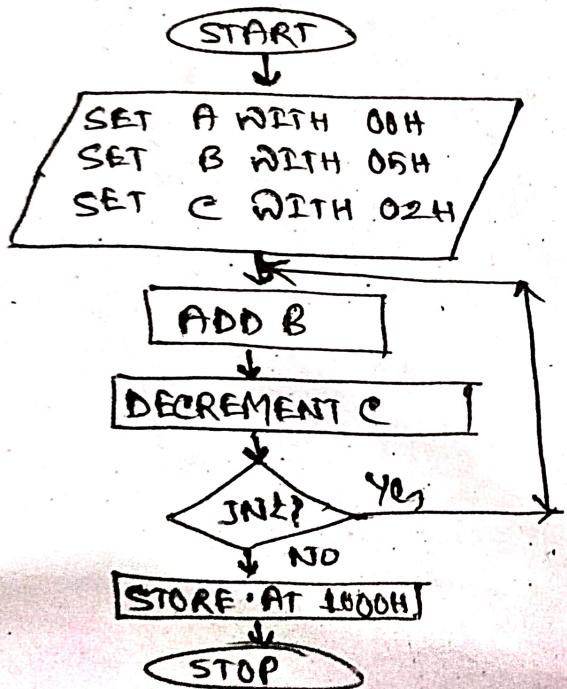
```

LDA 4000H
MOV B,A
ANI F0H
RRC
RRC
RRC
RRC
ADI 30H
STA 4001H
MOV A,B
ANI OFH
ADI 30H
STA 4002H
HLT.
  
```

STOP

11 Write a program to multiply two 8-bit numbers.

Flowchart:



Instruction:

MVI A,00H  
MVI B,05H  
MVI C,02H

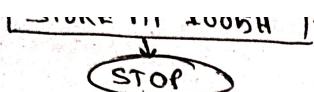
LOOP: ADD B

DEC C

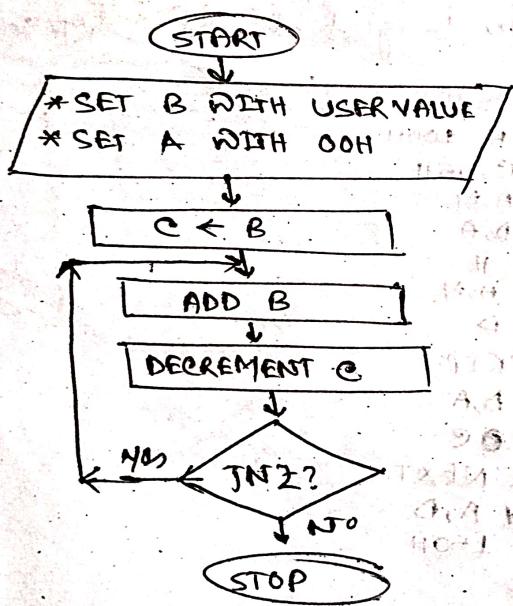
JNZ LOOP

STA 1000H

HLT.



Q3) Write an assembly level program to determine square of a 8bit number.  
 ↗ Flowchart:



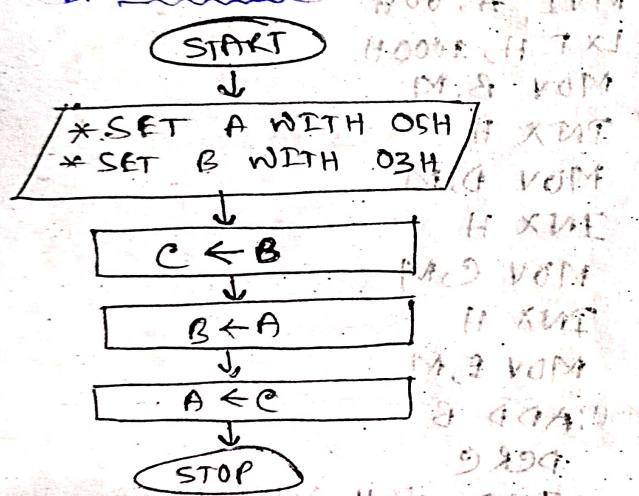
↗ Instruction:

```

MVI A,00H
MVI B,04H
MOV C,B
LOOP: ADD B
      DRC C
      JNZ LOOP
      HLT
  
```

Q14) Write an assembly level program to swap the content of two registers using any other general purpose register.

⇒ Flowchart:

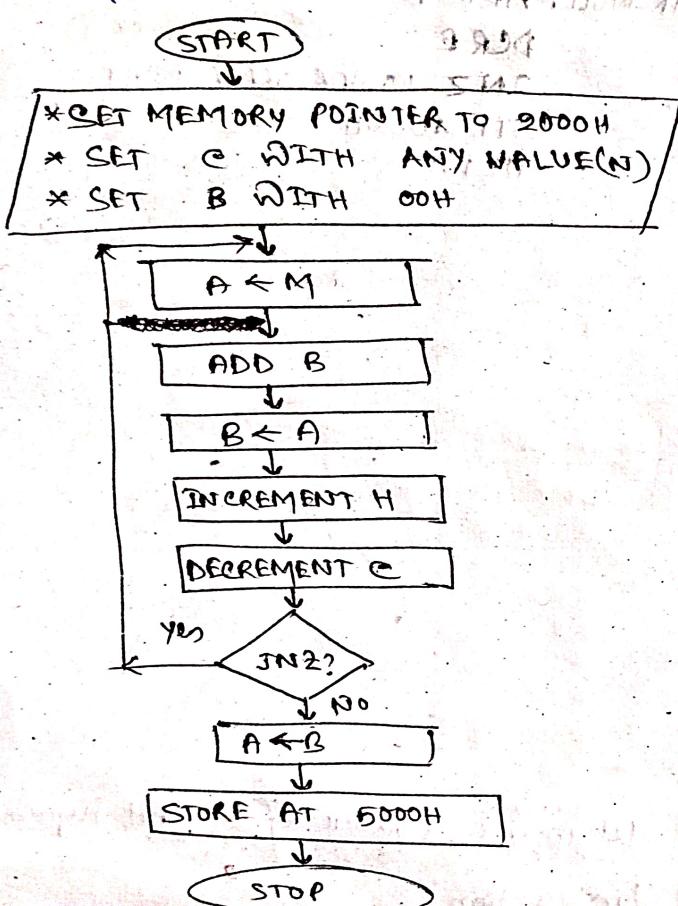


⇒ Instructions:

MVI A, 05H  
MVI B, 03H  
MOV C, B  
MOV B, A  
MOV A, C  
HLT.

Ques. Write an assembly level language program to add n numbers of 8-bit number stored in contiguous memory location.

Flowchart:



Instruction:

LXI H, 2000H

MVI C, 0AH

MVI B, 00H

LOOP:

MOV A, M

ADD B

MOV B, A

INX H

DEC C

JNZ LOOP

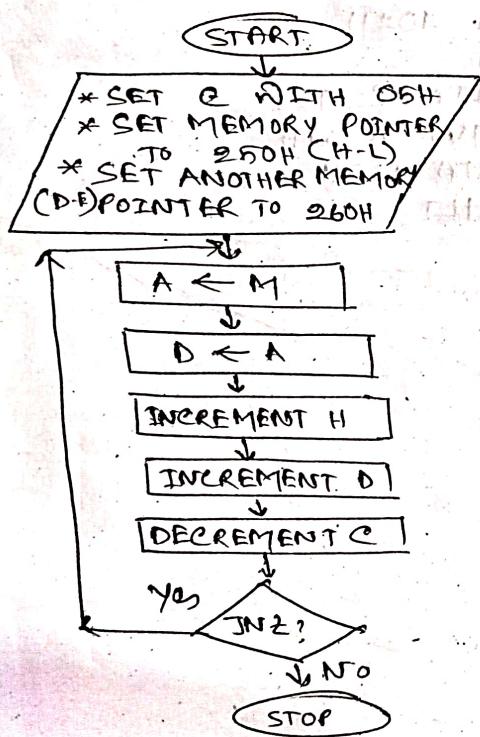
MOV A, B

STA 5000H

HLT.

19 Write an assembly level program to transfer block of data from one memory location to another.

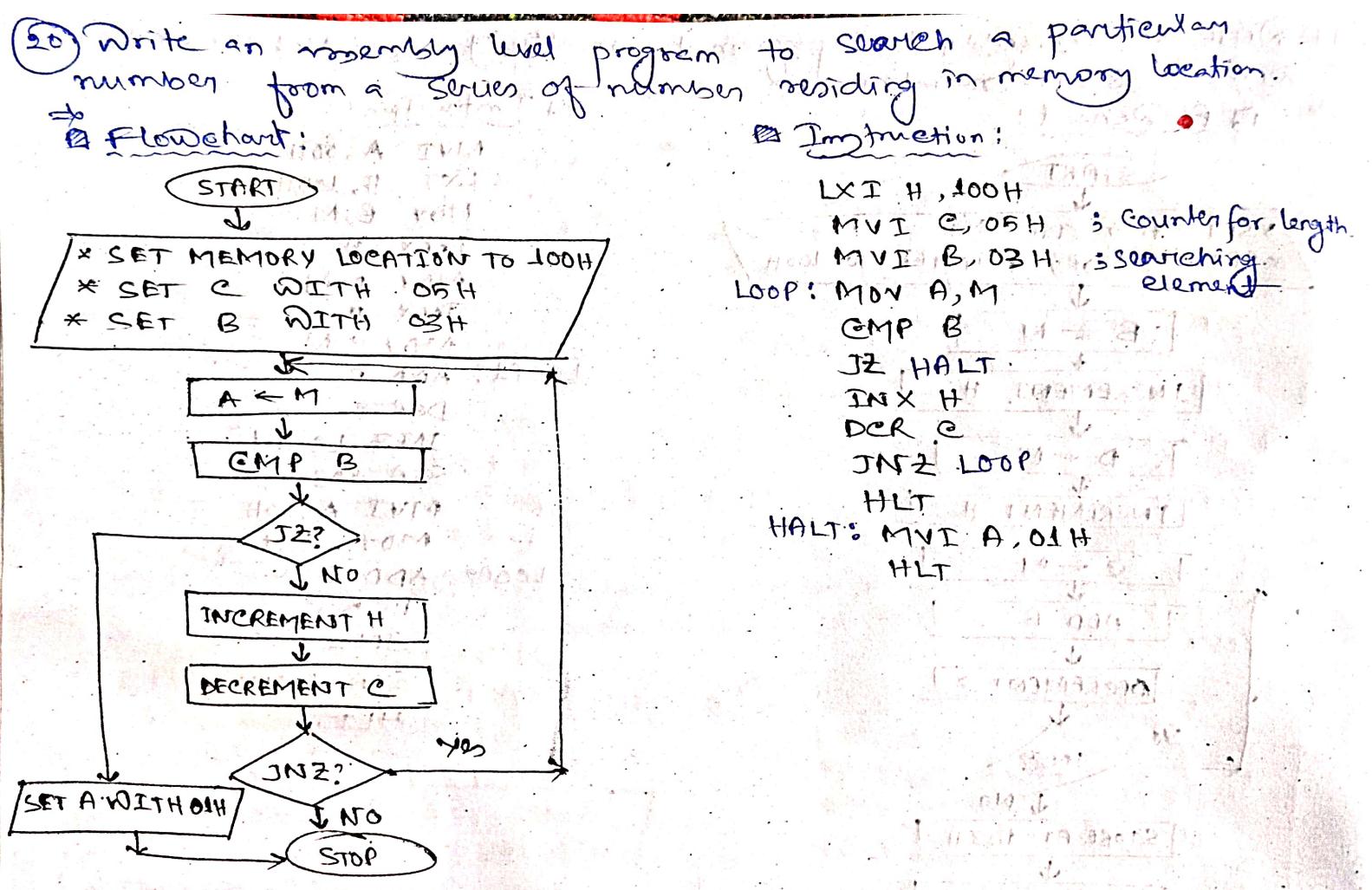
→ Flowchart:



Instruction:

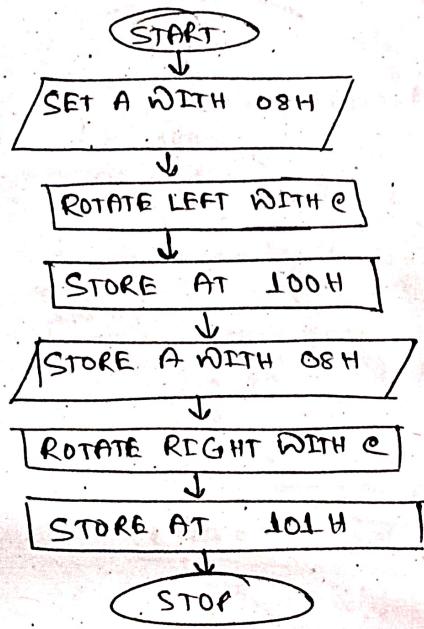
```

MVI C,05H
LXI H,250H
LXI D,260H
Loop: MOV A,M
STAX D
INX H
INX D
DCR C
JNZ Loop
HLT.
  
```



Q2 Write an assembly level program to illustrate multiplication by 2 and division by 2 by means of bit shifting operator.

\* Flowchart:



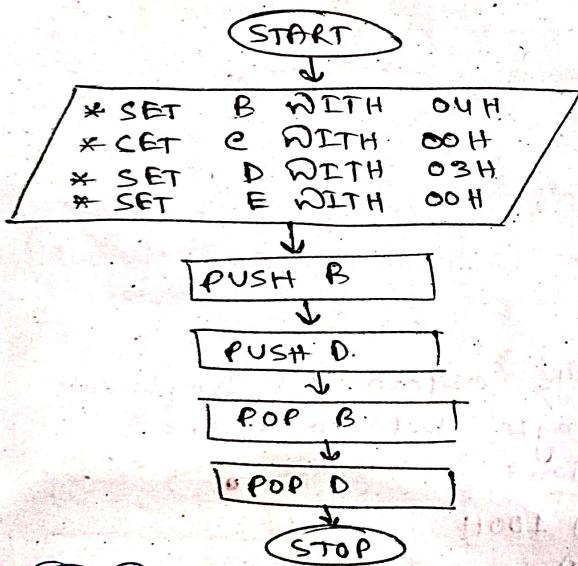
■ Instruction:

MVI A, 08H ;  
RLC  
STA 100H ; Multiplied by 2  
MVI A, 08H  
RRC  
STA 101H ; Divided by 2.  
HLT

STOP

24 Write an assembly level program to swap contents of two registers using stack pointer.

Flowchart:

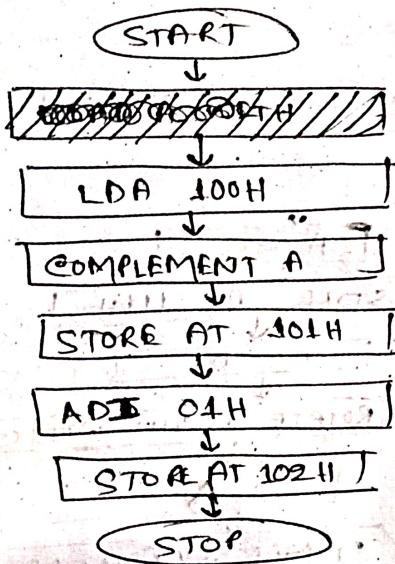


Instruction:

```
MVI B, 04H  
MVI C, 00H  
MVI D, 03H  
MVI E, 00H  
PUSH B  
PUSH D  
POP B  
POP D  
HLT
```

Q6) Write an assembly level language program to obtain 1's complement and 2's complement of a number ranges between 30 to 250.

→ Flowchart:

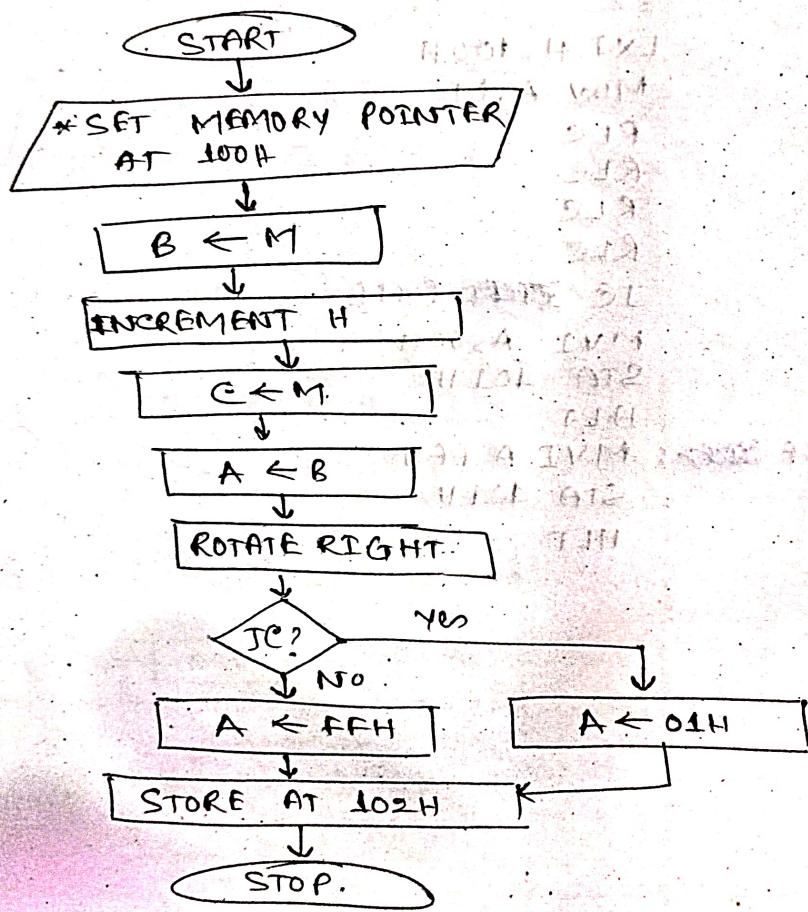


→ Instruction:

LDA 100H  
CMA  
STA 101H  
ADD 01H  
STA 102H  
HLT.

(43) Write an assembly level program to test whether a 16 bit number is odd or not.

⇒ Flowchart:



⇒ Instruction:

LXI H, 100H

MOV B, M

INX H

MOV C, M

MOV A, B

RRC

JC ODD

MVI A, FFH

STA 102H

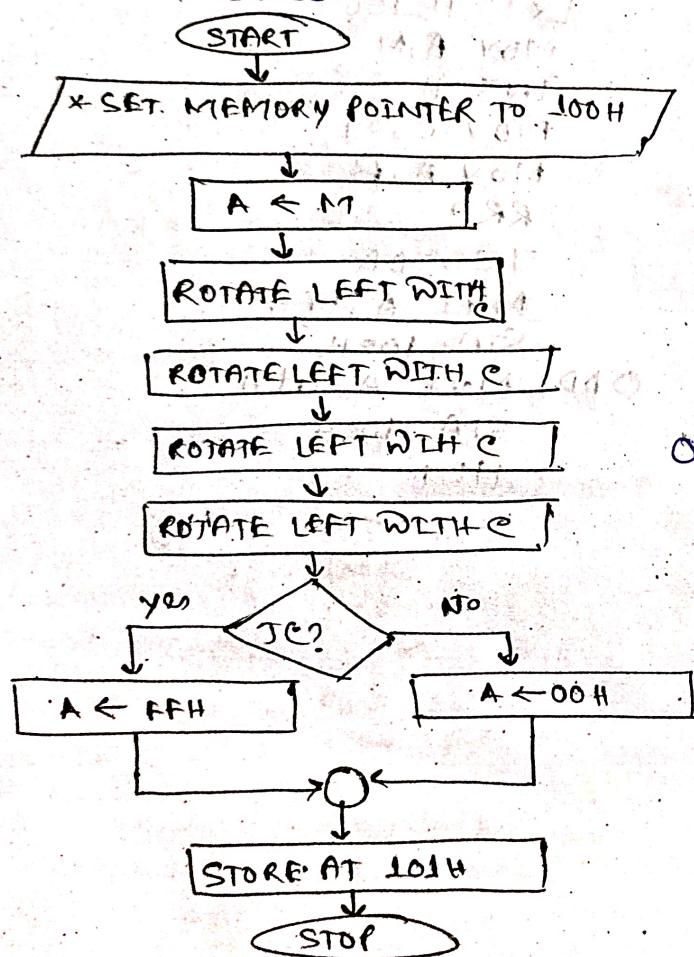
ODD: MVI A, 01H

STA 102H

HLT

② Write an assembly level program to check whether the 4th bit of the number is zero or one if 1 store FF otherwise 00.

→ Flowchart:



⇒ Instruction:

```
LXT H, 100H  
MOV A, M  
RLC  
RLC  
RLC  
RLC  
JC ZEROD, ONE  
MVI A, 00H  
STA 101H  
HLT  
ONE: MVI A, FFH  
STA 101H  
HLT.
```

↓ NO

STOP

- ⑦ Write an assembly level program to convert 8 bit data into BCD.

→ ~~Procedure~~: Instruction:

MVI A, 0EH

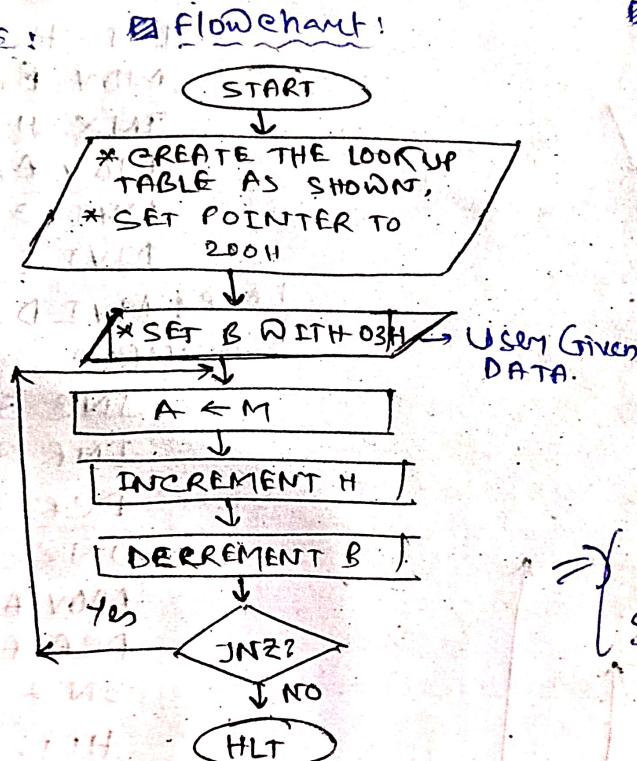
DAA

STA 100H ; 12 at 100H Memory location.

- ⑧ Write an assembly level program to reverse content of 10

Q) Write an assembly level language to find ~~cube~~<sup>cube root</sup> of a given number using look-up table.

200	$\sqrt[3]{1}$
201	$\sqrt[3]{2}$
202	$\sqrt[3]{3}$
203	$\sqrt[3]{4}$
204	$\sqrt[3]{5}$
205	$\sqrt[3]{6}$
:	



Q) Instruction:

```

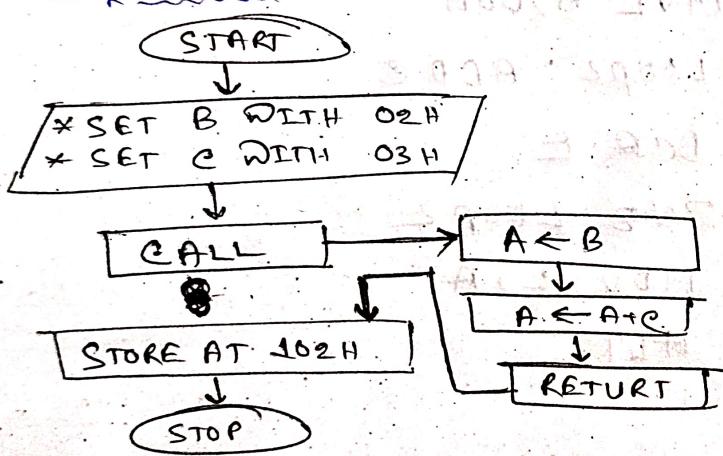
LXI H, 200H
MVI B, 03H ; User Given Data
LOOP: MOV A, M
       INX H
       DCR B
       JNZ LOOP
       HLT
  
```

4b  
Sub + reflect & swap  
BCD numbers

Q) Write an assembly level program

⑪ Write an assembly level program showing function of CALL instruction.

⇒ Flowchart:



Instruction:

LXI H, 100H

MOV B, M

INX H

MOV C, M

~~CALL~~

CALL SUM

STA 102H

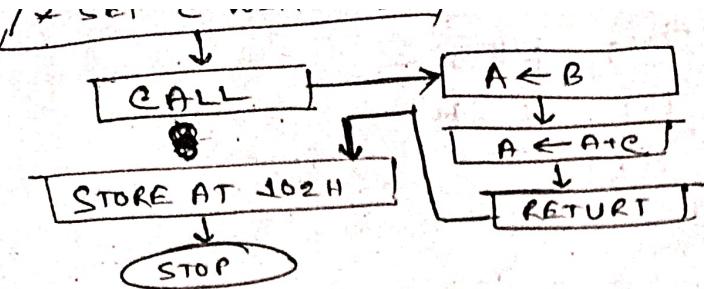
HLT

SUM:

MOV A, B

ADD C

RET



~~CALL SUM~~

STA 102H

HLT

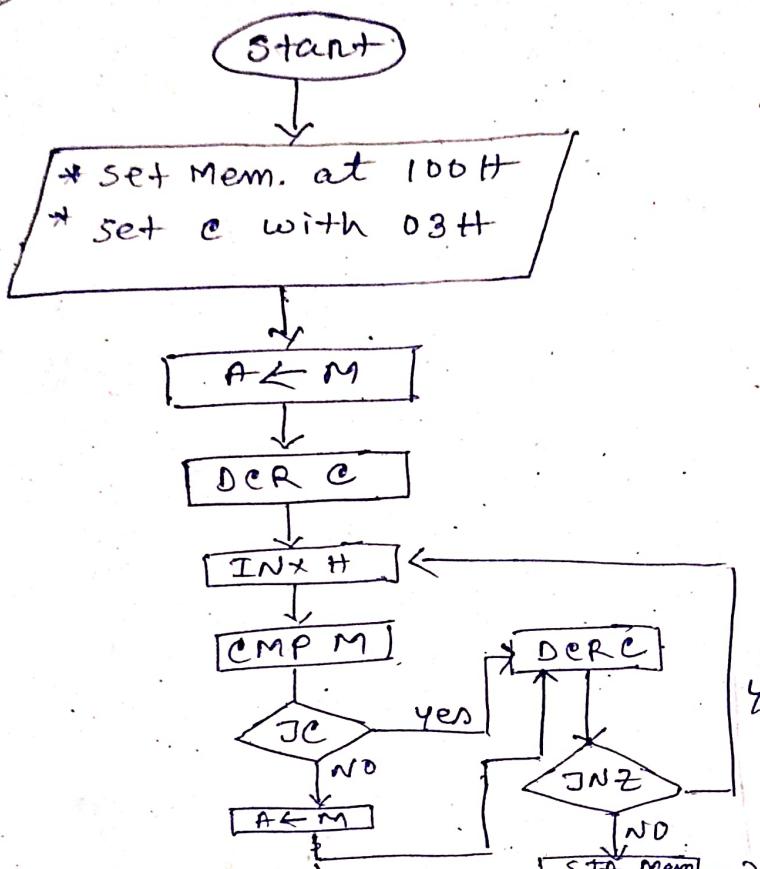
SUM:

MOVA,B

ADD C

RET

- (58) Find the lowest among five numbers. ①



Instruction:

LXI H, 100~~80~~H

MVI C, 05H

MOV A, M

DCR C

LOOP2: INX H

CMP M

JC LOOP1

MOV A, M

Loop1: DCR C

JNZ LOOP2

STA \$1040H

HLT

- (2) WAP to multiply two 16-bit numbers.

Sols.

LXI H, 100~~80~~H

MOV B, M

INX H

MOV C, M

INX H

MOV D, M

INX H

MOV E, M

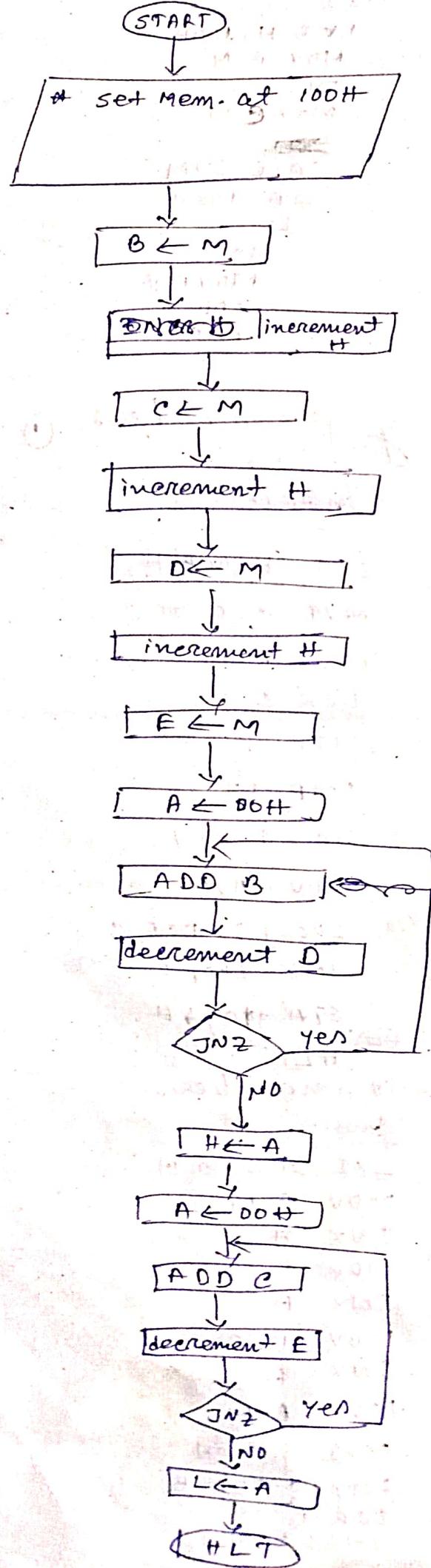
MVI A, 00H

Loop1: ADD B

DCR D

JNZ Loop1

## Flow chart:



MOV H, A

MVI A, 00H

LOOP2: ADD C

DCR E

JNZ LOOP2

MOV L, A

HLT