

Problem - 01

- Two numbers MN_H and KL_H are stored in 2050_H and 2051_H respectively.

Write a program to assemble them as NK_H and LM_H store them in 2052_H and 2053_H .

⇒ Algorithm :-)

LXI H, 2050_H

// Load the address in HL pair

MOV A, M

// Move the data to accumulator

ANI F0_H

// And with F0_H

RRC

// Rotate accumulator right

RRC

// Rotate Right

RRC

// Rotate Right

RRC

// Rotate Right

MOV D,A

// Copy Accumulator to D.

MOV A,M

// Move the data of M on A.

ANI OF_H

// AND with OF_H

RRC

// Rotate Accumulator right

RRC

// Rotate right

RRC

// Rotate right

RRC

// Rotate right .

MOV B,A

// Store Accumulator value to B.

LXI H, 2051_H

// Load the address in HL pair

MVI A, F0_H

// Move Immediate F0_H in Accumulator

ANA M

// And with the value at M

RRC

// Rotate Accumulator right

RRC

// Rotate right

RRC

// Rotate right

RRC // Rotate right
 ORA B // OR accumulator with B.
 STA 2052_H // Store Accumulator at 2052_H
 LXI H, 2051_H // Load address in HL register pair
 MVI A, 0F_H // Move 0F_H in Accumulator
 ANA M // And accumulator with value at M
 RRC // Rotate Right Accumulator
 RRC // Rotate right
 RRC // Rotate Right
 RRC // Rotate Right
 ORA D // OR accumulator with D.
 STA 2053_H // Store Accumulator at 2053_H
 HLT // Stop

Example :-

$$MN = AB \quad KL = CD$$

After execution,

$$NK = BC$$

$$\text{and} \quad LM = DA$$

Address	Command	Operation	Hex Code
2000	LXI H, 2050H	L ← 50H H ← 20H	21 50 20
2003	MOV A,M	A ← M[HL]	7E
2004	ANI F0H	A ← A & F0H	E6 F0
2006	RRC	Right Rotate	OF
2007	RRC	Rotate Right	OF
2008	RRC	Rotate Right	OF
2009	RRC	Rotate Right	OF
200A	MOV D,A	D ← A	5F
200B	MOV A,M	A ← M[HL]	7E
200C	ANI OFH	A ← A & OFH	E6 OF
200E	RRC	Rotate Right	OF
200F	RRC	Rotate Right	OF
2010	RRC	Rotate Right	OF
2011	RRC	Rotate Right	OF
2012	MOV B,A	B ← A	4F
2013	LXI H, 2051H	L ← 51H , H ← 20H	21 51 20
2016	MVI A, F0H	A ← F0H	3E F0
2018	ANA M	A ← A & M	46
2019	RRC	Rotate Right	OF
201A	RRC	Rotate Right	OF
201B	RRC	Rotate Right	OF
201C	RRC	Rotate Right	OF
201D	ORA B	A ← A OR B	B0
201E	STA 2052H	[2052] ← A	32 52 20

<u>Address</u>	<u>Command</u>	<u>Operation</u>	<u>Hex Code</u>
2021	LXI H, 2051H	$L \leftarrow 51_H, H \leftarrow 20_H$	21 51 20
2024	MVI A, OFH	$A \leftarrow OF_H$	3E OF
2026	ANA M	$A \leftarrow A \& M$	A6
2027	RRC	Rotate Right	OF
2028	RRC	Rotate Right	OF
2029	RRC	Rotate Right	OF
202A	RRC	Rotate Right	OF
202B	ORA D	$A \leftarrow A \text{ or } D$	B2
202C	STA 2053H	$[2053] \leftarrow A$	32 53 20
202F	HLT	Stop	76

— 0 —

Problem \Rightarrow 02

- Two numbers A and B are stored in 2050_H and 2051_H respectively.

Write a program to perform $A \times B$ and store the result in 2052_H and 2053_H .

\Rightarrow Algorithm \Rightarrow

LXI H, 2050_H // Load address in HL pair

MVI A, 00_H // Move 00 to Accumulator

MVI D, 00_H // Move 00 to D

ADD M // Add value of M and accumulator

JZ Loop3 // Jump to Loop3 if zero flag is 1

LXI H, 2051_H // Load address in HL pair.

MVI A, 00_H // Move 00 to Accumulator

ADD M // Add value of M and accumulator

JZ Loop3 // Jump to Loop3 if zero flag set

MOV E,A // Move A to E

LDA 2050_H // Load accumulator with 2050_H

MOV C,A // Move accumulator to C

MVI A, 00_H // Move 00 to accumulator

Loop1 : ADD E // Add accumulator with E

JNC Loop2 // Jump to Loop2 if carry is 0.

INR D // Increment D.

Loop2 : DCR C // Decrement C

JNZ Loop1 // Jump to Loop1 if zero flag is 0.

Loop3 : STA 2052_H // Store accumulator at 2052_H

MOV A,D // Move D to accumulator

STA 2053_H // Store A at 2053_H

HLT // Stop

Address	Command	Operation	HexCode
2000	LXI H, 2050H	L → 50, H ← 20	21 50 20
2003	MVI A, 00H	A ← 00H	3E 00
2005	MVI D, 00H	D ← 00H	16 00
2007	ADD M	A ← A + M[H:L]	86
2008	JZ 2024H	Jump if Zero	CA 24 20
200B	LXI H, 2051H	L → 51, H ← 20	21 51 20
200E	MVI A, 00H	A ← 00H	3E 00
2010	ADD M	A ← A + M[H:L]	86
2011	JZ 2024H	Jump if zero	CA 24 20
2014	MOV E,A	E ← A	5F
2015	LDA 2050H	A ← [2050]	3A 50 20
2018	MOV C,A	C ← A	4F
2019	MVI A,00H	A ← 00H	3E 00
201B	ADD E	A ← A + E	83
201E	JNC 2020H	Jump if NO Carry	D2 20 20
201F	INR D	D ← D + 1	14
2020	DCR C	C ← C - 1	0D
2021	JNZ 201BH	Jump not Zero	C2 1B 20
2024	STA 2052H	[2052] ← A	32 52 20
2027	MOV A,D	A ← D	7A
2028	STA 2053H	[2053] ← A	32 53 20
202B	HLT	Stop	76

Example ⇒ A = b8 B = d7 A × B = DA88

A = 67 B = d8 A × B = DA68

Problem - 03

N numbers are stored in consecutive m/m location starting from 2050_H . The value of N is stored in $204F_H$.

- (i) Find the maximum among the N numbers.

\Rightarrow	Address	Command	Operation	Hex Code
	2000	LDA $204F_H$	$A \leftarrow [204F]$	3A 4F 20
	2003	MOV D,A	$D \leftarrow A$	57
	2004	LXI H, 2050_H	$L \leftarrow 50, H \leftarrow 20$	21 50 20
	2007	MVI B, 00_H	$B \leftarrow 00_H$	06 00
	2009	MOV A,M	$A \leftarrow M[HL]$	7E
	200A	CMP B	Compare A with B	B8
	200B	JC $200F_H$	Jump if Carry	DA 0F 20
	200E	MOV B,A	$B \leftarrow A$	47
	200F	INX H	Increment HL	23
	2010	DCR D	$D \leftarrow D-1$	15
	2011	JNZ 2009_H	Jump if not zero	C2 09 20
	2014	MOV A,B	$A \leftarrow B$	78
	2015	STA 2150_H	$[2150] \leftarrow A$	32 50 21
	2018	HLT	Stop	76

— O —

Algorithm :-

```
LDA 204FH           // Load Accumulator from 204FH
MOV D,A              // Move Accumulator to D
LXI H,2050H         // Load address in HL pair
MVI B,00H           // Move 00 in B
Loop1: MOV A,M        // Move M[HL] in Accumulator
      CMP B            // Compare Accumulator with B
      JC Loop2          // If carry, then jump to Loop2
      MOV B,A           // Move Accumulator to B
Loop2: INX H           // Increment HL register pair
      DEC D             // Decrement D by 1
      JNZ Loop1         // If not zero, jump to Loop1
      MOV A,B           // Move B to accumulator.
      STA 2150H         // Store accumulator at 2150H
      HLT               // Stop
```

Example :-

$$N = 7$$

2050 _H	→ 21
2051 _H	→ b8
2052 _H	→ 53
2053 _H	→ F5
2054 _H	→ 76
2055 _H	→ C8
2056 _H	→ 76

AnsweR →

$$\textcircled{2} 150\text{H} \rightarrow F5$$

Problem - 03

- (ii) Find the minimum among the N numbers.

Address	Command	Operation	HexCode
2000	LDA 204F _H	A ← [204F]	3A 4F 20
2003	MOV D,A	D ← A	57
2004	LXI H, 2050 _H	L ← 50, H ← 20	21 50 20
2007	MVI B, FF _H	B ← FF _H	06 FF
2009	MOV A,M	A ← M[HL]	7E
200A	CMP B	Compare A with B	B8
200B	JNC 200F _H	Jump if no carry	D2 0F 20
200E	MOV B,A	B ← A	47
200F	INX H	Increment HL	23
2010	DCR D	D ← D-1	15
2011	JNZ 2009 _H	Jump if not zero	C2 09 20
2014	MOV A,B	A ← B	78
2015	STA 2150 _H	[2150] ← A	32 50 21
2018	HLT	Stop	76

— o —

Algorithm :-

```
LDA 204FH // Load accumulator from 204FH
MOV D,A // Move accumulator to D
LXI H, 2050H // Load address in HL pair.
MVI B, FF // Move FF in B
Loop1: MOV A,M // Move M[HL] in Accumulator
        CMP B // Compare accumulator with B
        JNC Loop2 // If not carry, then jump to loop2
        MOV B,A // Move accumulator to B
Loop2 : INX H // Increment HL register pair
        DCR D // Decrement D by 1
        JNZ Loop1 // If not zero, jump to loop1
        MOV A,B // Move B to accumulator
        STA 2150H // Store Accumulator at 2150H
        HLT // Stop
```

Example :-

$$N=7$$

2050 _H	→ b8
2051 _H	→ 21
2052 _H	→ 53
2053 _H	→ F5
2054 _H	→ 76
2055 _H	→ C8
2056 _H	→ 7b

Answer → 2150_H → 21

Problem - 03

(iii) Sort the N numbers in ascending Order.

Address	Command	Operation	Hex Code
2000	LXI H, 204FH	L \leftarrow 4FH, H \leftarrow 20H	21 4F 20
2003	MOV C, M	C \leftarrow M[HL]	4E
2004	DCR C	C \leftarrow C-1	0D
2005	MOV D, C	D \leftarrow C	51
2006	LXI H, 2050H	L \leftarrow 50H, H \leftarrow 20H	21 50 20
2009	MOV A, M	A \leftarrow M[HL]	7E
200A	INX H	Increment HL	23
200B	CMP M	Compare A with M	BE
200C	JC 2014H	Jump if carry	DA 14 20
200F	MOV B, M	B \leftarrow M[HL]	46
2010	MOV M, A	M[HL] \leftarrow A	77
2011	DCX H	Decrement HL	2B
2012	MOV M, B	M[HL] \leftarrow B	70
2013	INX H	Increment HL	23
2014	DCR D	D \leftarrow D-1	15
2015	JNZ 2009H	Jump if not zero	C2 09 20
2018	DCR C	C \leftarrow C-1	0D
2019	JNZ 2005H	Jump not zero	C2 05 20
201K	HLT	Stop	76

— —

Algorithm :-

LXI H, 204FH // Load address in HL pair
MOV C,M // Move M[HL] to C.
DCR C // Decrement C by 1
Loop1: MOV D,C // Move C to D
LXI H, 2050H // Load address in HL pair
Loop2 : MOV A,M // Move M[HL] to accumulator
INX H // Increment HL pair
CMP M // Compare Accumulator with M[HL]
JC Loop3 // Jump if carry to Loop3
MOV B,M // Move M[HL] to B
MOV M,A // Move accumulator to M[HL]
DCX H // Decrement HL pair
MOV M,B // Move B to M[HL]
INX H // Increment HL pair
Loop3: DCR D // Decrement D.
JNZ Loop2 // Jump if not zero to loop2
DCR C // Decrement C by 1
JNZ Loop1 // Jump if not zero Loop1
HLT // Stop

Example \Rightarrow

$$A = 7,$$

F5, b8, 85, 21, 76, C8, 7b

Ascending order \Rightarrow 21, 76, 7b, 85, b8, C8, F5

Problem - 03

(iv) Sort the N numbers in descending order.

=>	<u>Address</u>	<u>Command</u>	<u>Operation</u>	<u>Hex Code</u>
	2000	LXI H, 204FH	L ← 4F, H ← 20	21 4F 20
	2003	MOV C, M	C ← M[HL]	4E
	2004	DCR C	C ← C - 1	CD
	2005	MOV D, C	D ← C	51
	2006	LXI H, 2050H	L ← 50, H ← 20	21 50 20
	2009	MOV A, M	A ← M[HL]	7E
	200A	INX H	Increment HL	23
	200B	CMP M	Compare A with M	BE
	200C	JNC 2014H	Jump if not carry	D2 14 20
	200F	MOV B, M	B ← M[HL]	46
	2010	MOV M, A	M[HL] ← A	77
	2011	DCX H	Decrement HL	2B
	2012	MOV M, B	M[HL] ← B	70
	2013	INX H	Increment HL	23
	2014	DCR D	D ← D - 1	15
	2015	JNZ 2009H	Jump if not zero	C2 09 20
	2018	DCR C	C ← C - 1	0D
	2019	JNZ 2005H	Jump not zero	C2 05 20
	201C	HLT	Stop	76

Algorithm :-

LXI H, 204FH // Load address in HL pair
MOV C,M // Move M[HL] to C.
DCR C // Decrement C by 1
Loop1 : MOV D,C // Move C to D
LXI H, 2050H // Load address in HL pair
Loop2 : MOV A,M // Move M[HL] to Accumulator
INX H // Increment HL pair
CMP M // Compare accumulator with M[HL]
JMC Loop3 // Jump if not carry to loop3
MOV B,M // Move M[HL] to B
MOV M,A // Move accumulator to M[HL]
DCX H // Decrement HL pair
MOV M,B // Move B to M[HL]
INX H // Increment HL pair
Loop3 : DCR D // Decrement D by 1
JNZ Loop2 // Jump if not zero to loop2
DCR C // Decrement C by 1
JNZ Loop1 // Jump if not zero to loop1
HLT // Stop

Example ⇒

$$N = 7,$$

F5, b8, 85, 21, 76, C8, 7b

descending order ⇒ F5, C8, b8, 85, 7b, 76, 21

Problem - 04

= N numbers are stored in consecutive m/m location starting from 2050_H . The value of N is stored in $204F_H$. Write a program to copy the even and odd numbers starting from 2100_H and 2200_H respectively. Store the total number of even and odd numbers in 2300_H and 2301_H respectively.

Address	Command	Operation	Hex Code
2000	LDA $204F_H$	$A \leftarrow [204F]$	3A 4F 20
2003	MOV C, A	$C \leftarrow A$	4F
2004	LXI D, 2200_H	$E \leftarrow 00_H, D \leftarrow 22_H$	11 00 22
2007	MVI B, 00_H	$B \leftarrow 00_H$	06 00
2009	LXI H, 2050_H	$L \leftarrow 50, H \leftarrow 20$	21 50 20
$200C$	MOV A, M	$A \leftarrow M[HL]$	7E
$200D$	RAR	Rotate Right	1F
$200E$	JMC 2024_H	Jump not carry	D2 24 20
2011	RAL	Rotate Left	17
2012	INR B	$B \leftarrow B + 1$	04
2013	STAX D	$[DE] \leftarrow A$	12
2014	INX D	Increment DE	13
2015	JMP 2024_H	Jump not carry	C3 24 20
2018	MOV A, M	$A \leftarrow M[HL]$	7E
2019	RAR	Rotate right	1F
$201A$	JC $203C_H$	Jump if carry	DA 3C 20
$201D$	RAL	Rotate Left	17
$201E$	INR B	$B \leftarrow B + 1$	04

201F	STAX D	$[DE] \leftarrow A$	12
2020	INX D	Increment DE	13
2021	JMP 203CH	Jump to 203CH	C3 3C 20
2024	DCR C	$C \leftarrow C-1$	0D
2025	INX H	Increment HL	23
2026	JNZ 200CH	Jump not zero	C2 0C 20
2029	Mov A,B	$A \leftarrow B$	78
202A	STA 230H	$[230] \leftarrow A$	32 01 23
202D	LDA 204FH	$A \leftarrow [204F]$	3A 4F 20
2030	Mov C,A	$C \leftarrow A$	4F
2031	LXI H, 2050H	$L \leftarrow 50, H \leftarrow 20$	21 50 20
2034	LXI D, 2100H	$E \leftarrow 00, D \leftarrow 21$	11 00 21
2037	MVI B, 00H	$B \leftarrow 00H$	06 00
2039	JMP 2018H	Jump	C3 18 20
203C	DCR C	$C \leftarrow C-1$	0D
203D	INX H	Increment HL	23
203E	JNZ 2018H	Jump not zero	C2 18 20
2041	Mov A,B	$A \leftarrow B$	78
2042	STA 2300H	$[2300] \leftarrow A$	32 00 23
2045	HLT	Stop	76

Algorithm :-

LDA 204F_H // Load Accumulator from 204F_H

MOV C,A // Move A to C

LXI D, 2200_H // Load address in DE pair

MVI B, 00_H // Move 00 to B.

LXI H, 2050_H // Load address in HL pair

Loop1 : MOV A,M // Move M[HL] to A

RAR // Rotate A right

JNC Loop2 // Jump not carry to loop2

RAL // Rotate A left

INR B // Increment B

STAX D // Store accumulation in address of DE

INX D // Increment DE pair

JMP Loop2 // Jump to Loop2

Loop : MOV A,M // Move M[HL] to A

RAR // Rotate A right

JC Loop3 // Jump if carry to loop3

RAL // Rotate A left

INR B // Increment B

STAX D // Store accumulation in address in DE

INX D // Increment DE

JMP Loop3 // Jump to loop3

Loop2 : DCR C // Decrement C.

INX H // Increment HL pair

JNZ Loop1 // Jump not zero to loop1

MOV A,B // Move B to A

STA 2301_H // Store A in 2301_H

LDA 204F_H // Load A from 204F_H

MOV C,A // Move A to C

LXI H, 2050_H // Load address in HL pair

LXI D, 2100_H // Load address in DE pair

MVI B, 00_H // Move 00 to B

JMP Loop // Jump to loop

Loop3: DCR C // Decrement C.

INX H // Increment HL pair

JNZ Loop // Jump not zero to loop.

MOV A,B // Move B to A

STA 2300_H // Store A in 2300_H

HLT // Stop

Example :-

05, 02, 1B, 1A, 0A, 06, 09, AC

Odd \Rightarrow 05, 1B, 09

Even \Rightarrow 02, 1A, 0A, 06, ~~0~~, AC

Problem - 05

→ N numbers are stored in consecutive m/m location starting from 2050_H . The value N is stored in $204F_H$. Write a program to test whether a number stored in $204E_H$ is present in the list. If present, store its position in the list at $204D_H$, otherwise store FF_H .

⇒ Address	Command	Operation	Hex Code
2000	LDA $204E_H$	$A \leftarrow [204E]$	3A 4E 20
2003	MOV D, A	$D \leftarrow A$	57
2004	MVI B, 01 _H	$B \leftarrow 01_H$	05 01
2006	LDA $204F_H$	$A \leftarrow [204F]$	3A 4F 20
2009	MOV C, A	$C \leftarrow A$	4F
200A	LXI H, 2050 _H	$L \leftarrow 50, H \leftarrow 20$	21 50 20
200D	MOV A, M	$A \leftarrow M[HL]$	7E
200E	CMP D	Compare A with D	BA
200F	JNZ 2017_H	Jump not zero	C2 17 20
2012	MOV A, B	$A \leftarrow B$	78
2013	STA $204D_H$	$[204D] \leftarrow A$	32 4D 20
2016	HLT	Stop	76
2017	INR B	$B \leftarrow B+1$	04
2018	DCR C	$C \leftarrow C-1$	0D
2019	IMX H	Increment HL	23
201A	JNZ 2000_H	Jump not zero	C2 0D 20
201D	MVI A, FF _H	$A \leftarrow FF_H$	3E FF
201F	STA $204D_H$	$[204D] \leftarrow A$	32 4D 20
2022	HLT	Stop	76

Algorithm \Rightarrow

LDA 204E_H // Load A from 204E_H
MOV D,A // Move A to D
MVI B,01_H // Move D1 to B
LDA 204F_H // Load A from 204F_H
MOV C,A // Move A to C.

LXI H, 2050_H // Load address in HL pair

Loop: MOV A,M // Move M[HL] to A
CMP D // Compare A with D
JNZ Loop1 // Jump not zero to loop1
MOV A,B // Move B to A

STA 204D_H // Store A in 204D_H

HLT // Stop

Loop1 : INR B // Increment B by 1

DCR C // Decrement C

INX H // Increment HL pair

JNZ Loop // Jump not zero to loop

MVI A, FF_H // Move FF to A.

STA 204D_H // Store A in 204D_H

HLT // Stop.