MIT LAB 5

SAHIL BONDRE: U18CO021

1. A string of readings is stored in memory, locations starting at 2070H, and the end of the string is indicated by the byte 0DH.Write a program to check each byte in the string, and the save the bytes in the range of 30H to 39H (both inclusive) in memory locations starting from 2090H.

```
;<q1>
jmp start
;code
start: nop
lxi H, 2070H
lxi B, 2090H
loop: nop
mov A, M
xri 0DH
jz exit
mov A, M
cpi 30H
jc skip
cpi 39H
jz go
jnc skip
go: stax B
inx B
skip: nop
inx H
jmp loop
exit: hlt
```

Address (Hex)	Address	Data
2070	8304	48
2071	8305	12
2072	8306	78
2073	8307	19
2074	8308	50
2075	8309	13
2076	8310	0
2077	8311	0

Data 2070H onwards

Registers			Flag	
A	00		S	0
BC	20	92	Z	1
DE	08	00	AC	0
HL	20	75	P	1
PSW	00	00	C	0
PC	42	27		
SP	FF	FF		
Int-Reg	00			

Registers after execution

Address (Hex)	Address	Data
2090	8336	48
2091	8337	50
2092	8338	0
2093	8339	0
2094	8340	0
2095	8341	0
2096	8342	0
2097	8343	0

Copied Data 2090H onwards

2. A set of ten bytes is stored in memory starting with the address 2050H.Write a program to check each byte , and save the bytes that are higher than (60)10 and lower than (100)10 in memory locations starting from 2060H.

```
;<q2>
jmp start

start: nop
lxi H, 2050H
mvi D, 0AH
lxi B, 2060H
loop: nop
mov A, M
cpi 03CH
jc skip
cpi 64H
jnc skip
stax B
inx B
skip: nop
inx H
```

```
dcr D
jnz loop
hlt
```

Address (Hex)	Address	Data
2050	8272	78
2051	8273	12
2052	8274	45
2053	8275	84
2054	8276	21
2055	8277	59
2056	8278	64
2057	8279	100
2058	8280	33
2059	8281	1
205A	8282	0
205B	8283	0
205C	8284	0
205D	8285	0
205E	8286	0
205F	8287	0
2060	8288	78
2061	8289	84
2062	8290	64
2063	8291	0
2064	8292	0

Data from 2050H copied to 2060

Registers and Flags after execution

3. Data bytes are stored in memory locations from 2050H to 205FH.To insert an additional five bytes of data, It is necessary to shift the data string by five memory locations. Write a program to store the data string from 2055H to 2064H.Use any sixteen bytes of data to verify your program.

```
;<q3>

jmp start

;data
;code
start: nop
lxi H, 2050H
lxi B, 0010H
lxi D, 2055H
dad B
dcx H
xchg
dad B
dcx H
xchg
loop: nop
mov A, M
stax D
dcx D
dcx H
dcr C
jnz loop

hlt
```

Address (Hex)	Address	Data
2050	8272	45
2051	8273	120
2052	8274	11
2053	8275	7
2054	8276	2
2055	8277	15
2056	8278	11
2057	8279	17
2058	8280	17
2059	8281	78
205A	8282	86
205B	8283	71
205C	8284	26
205D	8285	21
205E	8286	59
205F	8287	69
2060	8288	0
2061	8289	0
2062	8290	0
2063	8291	0
2064	8292	0

Before

Address (Hex)	Address	Data
2050	8272	45
2051	8273	120
2052	8274	11
2053	8275	7
2054	8276	2
2055	8277	45
2056	8278	120
2057	8279	11
2058	8280	7
2059	8281	2
205A	8282	15
205B	8283	11
205C	8284	17
205D	8285	17
205E	8286	78
205F	8287	86
2060	8288	71
2061	8289	26
2062	8290	21
2063	8291	59
2064	8292	69

After

4. Write a Program to Sort the array in ascending order/descending order

```
;<q4>
jmp start

;data
name: db 85, 49, 56, 67, 79, 48, 50, 49

;code
start: lxi H, name
mvi D, 00H
; array size
mvi C, 8
dcr C
check: mov A, M
inx H
cmp M
; A <= [M]
jc next
jz next
; swap
mov B, M
mov M, A
dcx H
mov M, B
inx H
mvi D, 01H
next: dcr C
jnz check
mov A, D
cpi 01H
jz start
hlt
```

Address (Hex)	Address	Data
4203	16899	85
4204	16900	49
4205	16901	56
4206	16902	67
4207	16903	79
4208	16904	48
4209	16905	50
420A	16906	49

Unsorted data

Address (Hex)	Address	Data
4203	16899	48
4204	16900	49
4205	16901	49
4206	16902	50
4207	16903	56
4208	16904	67
4209	16905	79
420A	16906	85

Sorted data

5. Write a Program to find Largest number / Smallest number in a given data array.

```
;<q5>

jmp start

;code
start: nop
lxi H, 3000H
; counter
mvi C, 08H
mvi B, 0FFH
mvi D, 00H
loop: nop
mov A, M
cmp B
jnc skip
mov B, A
skip: nop
cmp D
jc skip2
mov D, A
skip2: nop
inx H
dcr C
jnz loop
hlt
```

Address (Hex)	Address	Data
3000	12288	12
3001	12289	45
3002	12290	11
3003	12291	78
3004	12292	99
3005	12293	5
3006	12294	65
3007	12295	27
3008	12296	0

The Given Data

Registers			Flag	
A	1B		S	0
BC	05	00	Z	1
DE	63	00	AC	0
HL	30	08	P	1
PSW	00	00	C	1
PC	42	21		
SP	FF	FF		
Int-Reg	00			

Min in B, Max in D

6. Write a Program to move a block starting at location 2000H To location 3000H With overlap/without overlap.

Without Overlap

```
;<q6a>

jmp start

;code
start: nop
lxi H, 3000H
mvi D, 05H
lxi B, 4000H
loop: nop
mov A, M
stax B
inx H
inx B
dcr D
jnz loop
```

```
hlt
```

Address (Hex)	Address	Data
3000	12288	1
3001	12289	2
3002	12290	3
3003	12291	4
3004	12292	5
3005	12293	0

Initial Data at 3000H

Address (Hex)	Address	Data
4000	16384	1
4001	16385	2
4002	16386	3
4003	16387	4
4004	16388	5
4005	16389	0
4006	16390	0

Copied Data to 4000H

With Overlap

```
;<q6-b>

jmp start

;code
start: nop
lxi H, 3000H
lxi B, 0006H
lxi D, 3004H
dad B
dcx H
xchg
dad B
dcx H
xchg
loop: nop
mov A, M
stax D
dcx D
dcx H
dcr C
```

```
jnz loop
hlt
```

Address (Hex)	Address	Data
3000	12288	1
3001	12289	2
3002	12290	3
3003	12291	4
3004	12292	5
3005	12293	6
3006	12294	0
3007	12295	0
3008	12296	0
3009	12297	0

Initial Data

Address (Hex)	Address	Data
3000	12288	1
3001	12289	2
3002	12290	3
3003	12291	4
3004	12292	1
3005	12293	2
3006	12294	3
3007	12295	4
3008	12296	5
3009	12297	6
300A	12298	0
300B	12299	0

After Copy