

Q2. Write a program to find smallest/largest number from the array of 8-bit data. Assume suitable memory location for data and result.

INPUT BEFORE EXECUTION

Data Stack KeyPad Memory I/O Ports		
Start	3000h	OK
Address (Hex)	Address	Data
3000	12288	10
3001	12289	1
3002	12290	8
3003	12291	6
3004	12292	22
3005	12293	9
3006	12294	2
3007	12295	5
3008	12296	8
3009	12297	3
300A	12298	0
300B	12299	0

AFTER EXECUTION

FINDING LARGEST ELEMENT OF ARRAY

Registers

A	16
BC	16 00
DE	00 00
HL	30 09
PSW	00 00
PC	42 1B
SP	FF FF
Int-Reg	00

Flag

S	0
Z	1
AC	0
P	1
C	1

Load me at

```

24
25 ;q2
26 ;Write a program to find smallest/largest number from the array of 8-bit
27 ;largest
28 mvi c,0Ah ; taking array size of 10
29 lxi h,3000h ;data from 3000h to 3009h
30 mov b,m
31 dcr c
32 loop: inx h
33 mov a,m
34 cmp b
35 jc skip ;if b>a then skip..(jump if carry)
36 mov b,a
37 skip: dcr c ;decrease count
38 jnz loop ;till 10 times
39 mov a,b
40 sta 3050h
41 hlt

```

Decimal - Hex Conversion

Decimal

Hex

0

0

→To Hex

←To Dec

Data Stack KeyPad Memory I/O Ports

Start 3050h OK

Address (Hex)	Address	Data
3050	12368	22
3051	12369	0
3052	12370	0
3053	12371	0
3054	12372	0
3055	12373	0
3056	12374	0
3057	12375	0
3058	12376	0
3059	12377	0
305A	12378	0
305B	12379	0

FINDING SMALLEST ELEMENT OF ARRAY

Registers

A	01
BC	01 00
DE	00 00
HL	30 09
PSW	00 00
PC	42 1B
SP	FF FF
Int-Reg	00

Flag

S	0
Z	1
AC	0
P	1
C	0

Load me at:

```

25 ;q2
26 ;Write a program to find smallest/largest number from the array of 8-bit
27
28
29
30 ;smallest
31 mvi c,0Ah ;array size of 10
32 lxi h,3000h
33 mov b,m
34 dcr c
35 loop: inx h
36 mov a,m
37 cmp b
38 jnc skip ;if b<=a then skip..(jump if not carry)
39 mov b,a
40 skip: dcr c ;decreas count
41 jnz loop ;till 10 times
42 mov a,b
43 sta 3050h
44 hlt
45
46
47
                
```

Data **Stack** **KeyPad** **Memory** **I/O Ports**

Start:

Address (Hex)	Address	Data
3050	12368	1
3051	12369	0
3052	12370	0
3053	12371	0
3054	12372	0
3055	12373	0
3056	12374	0
3057	12375	0
3058	12376	0
3059	12377	0
305A	12378	0
305B	12379	0

Line No Assembler Message

0 Program assembled successfully

Q3. Write a Program to search an 8-bit number from the array of 8-bit data.

INPUT BEFORE EXECUTION

Data **Stack** **KeyPad** **Memory** **I/O Ports**

Start:

Address (Hex)	Address	Data
3000	12288	22
3001	12289	1
3002	12290	8
3003	12291	6
3004	12292	22
3005	12293	9
3006	12294	2
3007	12295	5
3008	12296	8
3009	12297	3
300A	12298	19
300B	12299	0

Registers

A	04
BC	16 07
DE	00 00
HL	30 04
PSW	00 00
PC	42 21
SP	FF FF
Int-Reg	00

Flag

S	0
Z	0
AC	0
P	0
C	0

Decimal - Hex Conversion

Decimal

Hex

0

0

→ To Hex

← To Dec

I/O Ports

0

-

+

00

Update Port Value

Load me at

```

60
61 ;q3
62 ;Write a Program to search an 8-bit number from the array of 8-bit data
63
64 lda 3000h ;element to be found/searched
65 mov b,a
66 mvi c,0Ah ;array size of 10
67 lxi h,3001h ;array starting from 3001h to 300Ah
68 loop: mov a,m
69 cmp b ;a-b
70 jz skip ;if b=a then a-b=0 then skip..(jump if not carry)
71 inx h
72 dcr c
73 jnz loop ;till 10 times
74 jmp end ;if element not found in array
75 skip: mvi a,0Bh
76 sub c ;if element not found(c=0) 3050 is unchanged i.e, 0
77 sta 3050h ;array position considering 3001H to be position 1
78 end: hlt
79
80
81

```

Data Stack Keypad Memory I/O Ports

Start 3050h OK

Address (Hex)	Address	Data
3050	12368	4
3051	12369	0
3052	12370	0
3053	12371	0
3054	12372	0
3055	12373	0
3056	12374	0
3057	12375	0
3058	12376	0
3059	12377	0
305A	12378	0
305B	12379	0

Line No

Assembler Message

0 Program assembled successfully

PRINTS THE LOCATION OF ELEMENT IN ARRAY. CONSIDERING 3001 AS 1ST POSITION OF ARRAY.

Q4. Write a Program to arrange data in ascending/descending order. Assume suitable memory location for data and result.

INITIAL VALUE BEFORE EXECUTION

Data Stack Keypad Memory I/O Ports		
Start 3001h		OK
Address (Hex)	Address	Data
3001	12289	1
3002	12290	8
3003	12291	6
3004	12292	22
3005	12293	9
3006	12294	2
3007	12295	5
3008	12296	8
3009	12297	3
300A	12298	19
300B	12299	0
300C	12300	0

AFTER EXECUTION

ASCENDING ORDER

Registers

Register	Value	Flag
A	01	S 0
BC	03 00	Z 1
DE	00 00	AC 0
HL	30 02	P 1
PSW	00 00	C 1
PC	42 1F	
SP	FF FF	
Int-Reg	00	

Decimal - Hex Conversion

Decimal	Hex
0	0

I/O Ports

Port	Value
0	00

Code

```

82
83 ;Write a Program to arrange data in ascending/descending order.
84 ;ascending order
85 MVI C,0AH ;LOAD COUNT FO ARRAY =10
86 DCR C ;Decrement Count
87 DO: MOV D,C
88 LXI H,3001h ;array starts from 3001h to 300Ah
89 LOOP: MOV A,M
90 INX H
91 CMP M
92 JC SKIP ;jump to skip if carry generated
93 MOV B,M ;swap memry and accumalator data if carry =0
94 MOV M,A
95 DCX H
96 MOV M,B
97 INX H
98 SKIP: DCR D
99 JNZ LOOP ;jump to loop if not zero
100 DCR C
101 JNZ DO
102 HLT
103

```

Memory

Address (Hex)	Address	Data
3001	12289	1
3002	12290	2
3003	12291	3
3004	12292	5
3005	12293	6
3006	12294	8
3007	12295	8
3008	12296	9
3009	12297	19
300A	12298	22
300B	12299	0
300C	12300	0

Assembler Message

Line No	Assembler Message
0	Program assembled successfully

DESCENDING ORDER

Registers

Register	Value	Flag
A	13	S 0
BC	16 00	Z 1
DE	00 00	AC 0
HL	30 02	P 1
PSW	00 00	C 1
PC	42 1F	
SP	FF FF	
Int-Reg	00	

Decimal - Hex Conversion

Decimal	Hex
0	0

I/O Ports

Port	Value
0	00

Code

```

81 ;q4
82
83 ;Write a Program to arrange data in ascending/descending order.
84 ;descending order
85 MVI C,0AH ;LOAD COUNT FO ARRAY =10
86 DCR C ;Decrement Count
87 DO: MOV D,C
88 LXI H,3001h ;array starts from 3001h to 300Ah
89 LOOP: MOV A,M
90 INX H
91 CMP M
92 JNC SKIP ;jump to skip if carry not generated
93 MOV B,M ;swap memry and accumalator data
94 MOV M,A
95 DCX H
96 MOV M,B
97 INX H
98 SKIP: DCR D
99 JNZ LOOP ;jump to loop if not zero
100 DCR C
101 JNZ DO
102 HLT
103

```

Memory

Address (Hex)	Address	Data
3001	12289	22
3002	12290	19
3003	12291	9
3004	12292	8
3005	12293	8
3006	12294	6
3007	12295	5
3008	12296	3
3009	12297	2
300A	12298	1
300B	12299	0
300C	12300	0

Assembler Message

Line No	Assembler Message
0	Program assembled successfully