

Name : Khushi Nitinkumar Patel

PRN : 2020BTECS00037

EXPERIMENT NO 11

Title of experiment : WAP for arranging numbers in ascending and descending order.

Equipment required : GNU Simulator.

Theory :

For ascending order :

- 1) Initialize HL pair as memory pointer.
- 2) Get the count at 4200 in to C register.
- 3) Copy it in D register.
- 4) Get the first value in Accumulator.
- 5) Compare it with the value at next location.
- 6) If they are out of order, exchange the contents of accumulator and memory.
- 7) Decrement D register's content by 1.
- 8) Repeat steps 5 and 7 till the value in D register become zero.
- 9) Decrement C register's content by 1.
- 10) Repeat steps 3 to 9 till the value in C register becomes zero.
- 11) Terminate the program.

For descending :

- 1)Initialize HL pair as memory pointer.
- 2)Get the count at 4200 in to C register.
- 3)Copy it in D register.
- 4)Get the first vale in Accumulator.
- 5)Compare it with the value at next location.
- 6)If they are out of order, exchange the contents of accumulator and memory.
- 7)Decrement D register's content by 1.
- 8)Repeat steps 5 and 7 till the value in D register become zero.
- 9)Decrement C register's content by 1.
- 10)Repeat steps 3 to 9 till the value in C register becomes zero.
- 11)Terminate the program.

Program code :

;Arrange in ascending order

LDA 1100

MOV B,A

DCR B

LOOP3: LXI H,1100

MOV C,M

DCR C

INX H

LOOP2: MOV A,M

INX H

CMP M

JC LOOP1

MOV D,M

MOV M,A

DCX H

MOV M,D

INX H

LOOP1: DCR C

JNZ LOOP2

DCR B

JNZ LOOP3

HLT

;Arrange in descending order

LDA 1100

MOV B,A

DCR B

LOOP3: LXI H,1100

MOV C,M

DCR C

INX H

LOOP2: MOV A,M

INX H

CMP M

JNC LOOP1

MOV D,M

MOV M,A

DCX H

MOV M,D

INX H

LOOP1: DCR C

JNZ LOOP2

DCR B

JNZ LOOP3

HLT

Snap shots :

Arrange in ascending order :

The screenshot shows the GNUSim8085 - 8085 Microprocessor Simulator interface. The main window displays the assembly code and the state of the microprocessor. The registers are shown on the left, and the memory is shown on the right. The assembly code is as follows:

```
1
2 ;Arrange in ascending order
3
4 LDA 1100
5 MOV B,A
6 DCR B
7 LOOP3: LXI H,1100
8 MOV C,M
9 DCR C
10 INX H
11 LOOP2: MOV A,M
12 INX H
13 CMP M
14 JC LOOP1
15 MOV D,M
16 MOV M,A
17 DCX H
18 MOV M,D
19 INX H
20 LOOP1: DCR C
21 JNZ LOOP2
22 DCR B
23 JNZ LOOP3
24 HLT
25
```

The registers are shown on the left, and the memory is shown on the right. The memory table is as follows:

Address (Hex)	Address	Data
044C	1100	5
044D	1101	23
044E	1102	44
044F	1103	12
0450	1104	6
0451	1105	33
0452	1106	0
0453	1107	0
0454	1108	0
0455	1109	0
0456	1110	0
0457	1111	0

The simulator status is "Idle".

The screenshot shows the GNUSim8085 - 8085 Microprocessor Simulator interface after execution. The registers are shown on the left, and the memory is shown on the right. The assembly code is the same as in the first screenshot. The registers are shown on the left, and the memory is shown on the right. The memory table is as follows:

Address (Hex)	Address	Data
044C	1100	5
044D	1101	6
044E	1102	12
044F	1103	23
0450	1104	33
0451	1105	44
0452	1106	0
0453	1107	0
0454	1108	0
0455	1109	0
0456	1110	0
0457	1111	0

The simulator status is "Idle".

Arrange in descending order :

The screenshot shows the GNUSim8085 - 8085 Microprocessor Simulator interface. The assembly code in the center pane is as follows:

```
1
2 ;Arrange in descending order
3 LDA 1100
4 MOV B,A
5 DCR B
6 LOOP3: LXI H,1100
7 MOV C,M
8 DCR C
9 INX H
10 LOOP2: MOV A,M
11 INX H
12 CMP M
13 JNC LOOP1
14 MOV D,M
15 MOV M,A
16 DCX H
17 MOV M,D
18 INX H
19 LOOP1: DCR C
20 JNZ LOOP2
21 DCR B
22 JNZ LOOP3
23 HLT
24
25
26
```

The registers on the left show: A=21, BC=00 00, DE=06 00, HL=04 51, PSW=00 00, PC=42 1F, SP=FF FF, Int-Reg=00. The flags on the right show: S=0, Z=1, AC=0, P=1, C=1. The memory window on the right shows the following data:

Address (Hex)	Address	Data
044C	1100	5
044D	1101	6
044E	1102	12
044F	1103	23
0450	1104	33
0451	1105	44
0452	1106	0
0453	1107	0
0454	1108	0
0455	1109	0
0456	1110	0
0457	1111	0

The assembler message at the bottom states: "Program assembled successfully".

The screenshot shows the GNUSim8085 - 8085 Microprocessor Simulator interface. The assembly code in the center pane is as follows:

```
1
2 ;Arrange in descending order
3 LDA 1100
4 MOV B,A
5 DCR B
6 LOOP3: LXI H,1100
7 MOV C,M
8 DCR C
9 INX H
10 LOOP2: MOV A,M
11 INX H
12 CMP M
13 JNC LOOP1
14 MOV D,M
15 MOV M,A
16 DCX H
17 MOV M,D
18 INX H
19 LOOP1: DCR C
20 JNZ LOOP2
21 DCR B
22 JNZ LOOP3
23 HLT
24
25
26
```

The registers on the left show: A=0C, BC=00 00, DE=2C 00, HL=04 51, PSW=00 00, PC=42 1F, SP=FF FF, Int-Reg=00. The flags on the right show: S=0, Z=1, AC=0, P=1, C=0. The memory window on the right shows the following data:

Address (Hex)	Address	Data
044C	1100	5
044D	1101	44
044E	1102	33
044F	1103	23
0450	1104	12
0451	1105	6
0452	1106	0
0453	1107	0
0454	1108	0
0455	1109	0
0456	1110	0
0457	1111	0

The assembler message at the bottom states: "Program assembled successfully".

Conclusion : These are ways to perform ascending and descending order