Name: Jigar Siddhpura **SAPID:** 60004200155

DIV: C/C2 **Branch:** Computer Engineering

POA EXPERIMENT 8

60004210155
Jigar Siddhpura

Experiment 8 : Maximum / Minimum no. in Array

Aim : Implement max / min. no. from a array in assembly program

Theory : 1. Experiment is about finding max & min. no. from an array.

2. For which ALP program utilizes the REPEAT loop instr. to iterate through each element of the array, comparing current element with max & min. element previously determined.

3. Firstly the declaration & initialization of array named ARR with set of integers LEN, MIN, MAX.

4. CHECK MAX is part of conditional logic.

5. After comparing the current element with min. value, it checks whether it is greater than max value.

6. If Yes, program jumps to CHECK MAX to update max value ensuring both min. & max. are updated in each iteration properly.

7. If current element is smaller than ~~current~~, min. recorded value & is greater than ~~gw~~ current max. ; max value is updated.

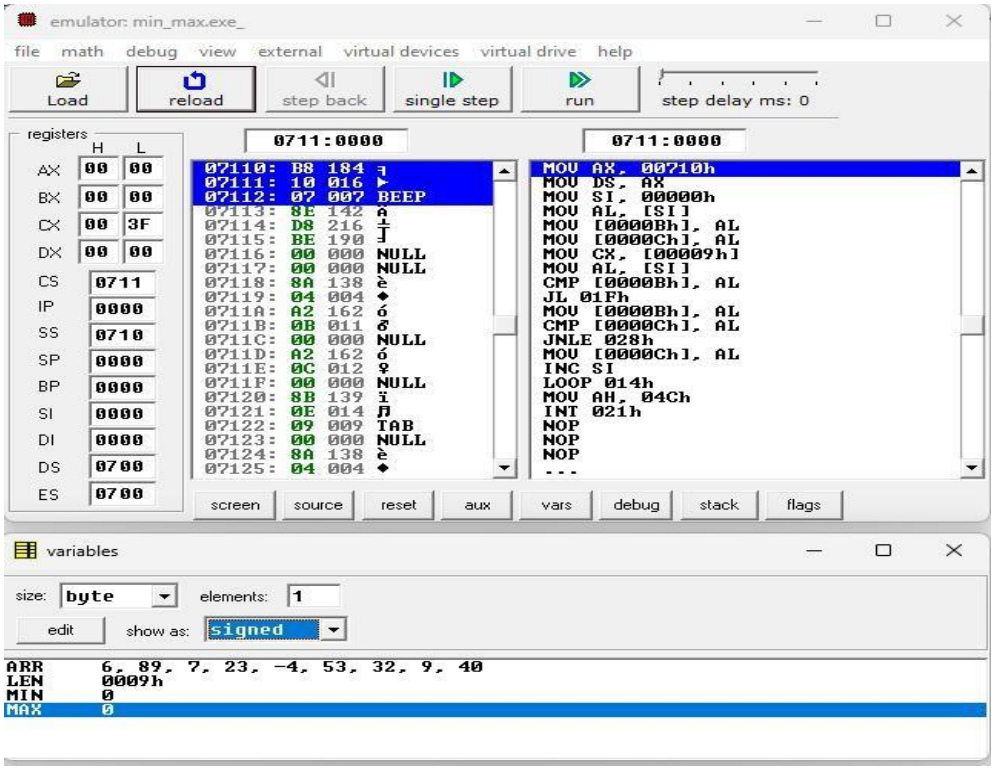
8. Here MOV, LEA, CMP, JL, JG (Jump if Less & Jump if greater), LOOP, INT21 H instr are used

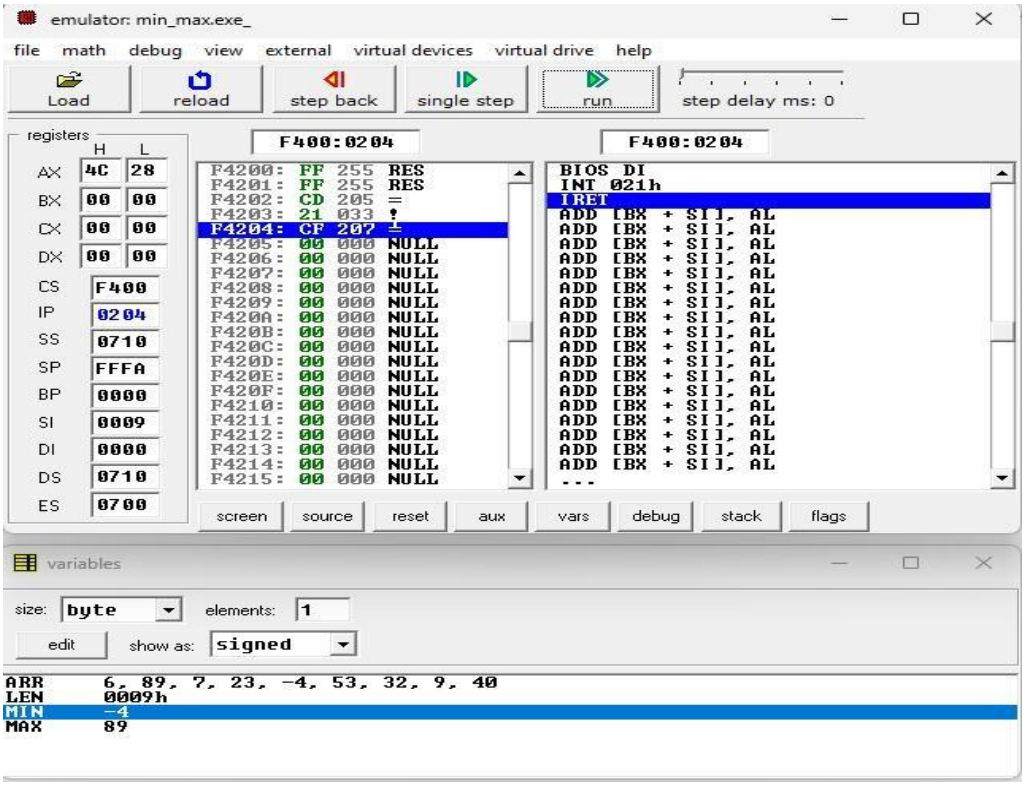
Conclusion : Hence, we implemented assembly program to find max. & min. from an array.

Code :

```
org 100h

DATA SEGMENT
   ARR DB 6,89,7,23,-4,53,32,9,40
   LEN DW SI $-ARR
   MIN DB ?
   MAX DB ?
   DATA ENDS

CODE SEGMENT
   START:
   MOV AX,DATA
   MOV DS,AX
   LEA SI,ARR
   MOV AL,ARR[SI]
   MOV MIN,AL
   MOV MAX,AL
   MOV CX,LEN

   REPEAT:
   MOV AL,ARR[SI]
   CMP MIN,AL
   JL CHECKMAX
   MOV MIN,AL
   CHECKMAX:
   CMP MAX,AL
   JG DONE
   MOV MAX,AL
   DONE:
   INC SI
   LOOP REPEAT


   MOV AH,4CH
   INT 21H

   CODE ENDS
END START

Ret
```

Output :


emulator: min_max.exe_
file math debug view external virtual devices virtual drive help
Load reload step back single step run step delay ms: 0
registers
H L
AX 00 00
BX 00 00
CX 00 3F
DX 00 00
CS 0711
IP 0000
SS 0710
SP 0000
BP 0000
SI 0000
DI 0000
DS 0700
ES 0700
0711:0000
0711:0000
07110: B8 184
07111: 10 016
07112: 07 007 BEEP
07113: 8E 142
07114: D8 216
07115: BE 190
07116: 00 000 NULL
07117: 00 000 NULL
07118: 8A 138
07119: 04 004
0711A: A2 162
0711B: 0B 011
0711C: 00 000 NULL
0711D: A2 162
0711E: 0C 012
0711F: 00 000 NULL
07120: 8B 139
07121: 0E 014
07122: 09 009 TAB
07123: 00 000 NULL
07124: 8A 138
07125: 04 004
MOV AX, 00710h
MOV DS, AX
MOV SI, 00000h
MOV AL, [SI]
MOV [0000Bh], AL
MOV [0000Ch], AL
MOV CX, [00009h]
MOV AL, [SI]
CMP [0000Bh], AL
JL 01Fh
MOV [0000Bh], AL
CMP [0000Ch], AL
JNLE 028h
MOV [0000Ch], AL
INC SI
LOOP 014h
MOV AH, 04Ch
INT 021h
NOP
NOP
NOP
screen source reset aux vars debug stack flags
variables
size: byte elements: 1
edit show as: signed
ARR 6, 89, 7, 23, -4, 53, 32, 9, 40
LEN 0009h
MIN 0
MAX 0


emulator: min_max.exe_
file math debug view external virtual devices virtual drive help
Load reload step back single step run step delay ms: 0
registers
H L
AX 4C 28
BX 00 00
CX 00 00
DX 00 00
CS F400
IP 0204
SS 0710
SP FFFA
BP 0000
SI 0009
DI 0000
DS 0710
ES 0700
F400:0204
F400:0204
F4200: FF 255 RES
F4201: FF 255 RES
F4202: CD 205 =
F4203: 21 033 !
F4204: CF 207 ╧
F4205: 00 000 NULL
F4206: 00 000 NULL
F4207: 00 000 NULL
F4208: 00 000 NULL
F4209: 00 000 NULL
F420A: 00 000 NULL
F420B: 00 000 NULL
F420C: 00 000 NULL
F420D: 00 000 NULL
F420E: 00 000 NULL
F420F: 00 000 NULL
F4210: 00 000 NULL
F4211: 00 000 NULL
F4212: 00 000 NULL
F4213: 00 000 NULL
F4214: 00 000 NULL
F4215: 00 000 NULL
BIOS DI
INT 021h
IRET
ADD [BX + SI], AL
screen source reset aux vars debug stack flags
variables
size: byte elements: 1
edit show as: signed
ARR 6, 89, 7, 23, -4, 53, 32, 9, 40
LEN 0009h
MIN -4
MAX 89