

EXPERIMENT NO- 7

AIM: WAP to find Maximum number from an array

Resource Required: P-IV and above RAM 128MB, Dot Matrix Printer, Emu 8086, MASM 611/ TASM, Turbo C/C++, Printer, Printout Stationary.

THEORY:

Instructions used in this program are:

- 1) **LEA:** Load Effective Address loads the specified register with the offset of a memory location.

Syntax: LEA register, memory

- 2) **JC (Jump if carry):** This is conditional Jump. This instruction will jump to specified label when carry flag is set.

- 3) **JNZ (Jump if not Zero):** This is conditional Jump. This instruction will jump to specified label when zero flag is not set.

- 4) **DEC:** DEC decrements the source by one

Syntax: DEC source

- 5) **CMP:** Compare the numerical value of the destination with the source and set flags appropriately. This comparison is carried out in the form of a subtraction to determine which of the operands has a greater value. After a CMP instruction, OF, SF, ZF and CF are set appropriately. For example, if the operands have equal values, then ZF is set.

Syntax:

CMP destination, source

ALGORITHM:

- | | |
|-----------------|---------------------------------|
| Step I | : Initialize the data segment |
| Step II | : Move the length to CX |
| Step III | : Initialize BL register to 00h |

- Step IV** : Load the first number to SI register.
- Step V** : Move contents pointed by SI register to AL register
- Step VI** : Compare the contents of AL and BL register.
- Step VII** : If $AL < BL$, then go to Step IX else go to Step VIII
- Step VIII** : Move the contents of AL to BL register
- Step IX** : Increment SI
- Step X** : Decrement CX contents by 1
- Step XI** : Repeat till $CX=0$
- Step XII** : Display the value in max variable
- Step XIII** : Stop

CONCLUSION: We have successfully found the maximum number from an array in Assembly Language using EMU8086.

Code and Output:

The screenshot displays the EMU8086 emulator interface with the following components:

- Source Code Window:** Shows the assembly code for finding the maximum number in an array.


```

01 Data segment
02     arr db 95h,98h,69h,55h,99h
03     max db ?
04 Data ends
05
06 Code segment
07 Assume ds:Data cs:Code
08
09 Start:
10     Mov ax,Data
11     Mov ds,ax
12     Lea si,arr
13     Mov cx,05h
14     Mov bl,00h
15
16 Up:
17     Mov al,[si]
18     Cmp al,bl
19     Jc Down
20     Mov bl,al
21     Inc si
22     Dec cx
23     Jnz Up
24
25 Int 03h
26 Code ends
27 END Start
      
```
- Registers Window:** Displays the state of 8086 registers.

Register	Value
AX	07 99
BX	00 99
CX	00 00
DX	00 00
CS	F400
IP	0104
SS	0710
SP	FFFA
BP	0000
SI	0005
DI	0000
DS	0710
ES	0700
- Variables Window:** Shows the memory locations for the array and the maximum value.

Variable	Value
ARR	95h, 98h, 69h, 55h, 99h
MAX	99h
- Flags Window:** Displays the status of various flags.

Flag	Value
CF	0
ZF	1
SF	0
OF	0
PF	1
AF	0
IF	0
DF	0
- Original Source Window:** Shows the original source code with syntax highlighting.