-	I

Nance: Manthan . M. Sonawane

Panel: E Poll No: 17

MAIOT LAB Assignment -11

Problem Statement: Write an ALP to sort 8-bit numbers in ascending or descending order

Theory: Explain new instructions used

- 1. ROL used to rotate bik of byse word towards left.
- 2. AND used for adding each bit in a byte / word with the corresponding bit in another byte | word
- 3. CMP- used to compose 2 provided byte /word
- 4. JBE used to pump jump before instruction satisfies
- S. JNZ- wed to jump it net zero flag zf=0
- 6. JAE used to jump if above instructions satisfies

Algorith used

- 1) For sorting the integer numbers in ascending descending
 - 1. Declare numbers to be sorted
 - 2. Specify counter to perform sorting
- 3. Select pointer
 - 4. Decide instruction to be used for decisions of greatest smaller number
 - s. Arrange the sorted in ascending or descending order.
 - 6. Use 2 digit display for unpacking of numbers
 - 7. Print all numbers and terminate the code

	Page No. Date
	Display sorted numbers
	1) set at +05
	2. Store array in 181
	B. Store result in rdi
8.636	4. Set du to 2
	5. Store (rsi] in al
	6. Retate at 4 times to the left
	7. Store at in bl
19	8. And al and Fh
	9. compare al with gH
	10. If less than 9H go to step 12
	The ADD at and 7H
	12. ADD 30H to al
	13. dectal reduced your grown area of hour - 387.
	14. If not 0, go to step 6
	15. Store AH en rili
	16- increment rdi
	17 · decrement a
600	18. 1Fnot 0, step 4
	: tame
#	Platform -> open source linux
	2. Specifica rountes to peations sorting
#	Conclusion: Thus, the program is implemented in an assembly language to sort 8 bit number
1	assembly language to sort 8 bit number
	cadowa sallone
#	FAG'S no enthances to enthance at below all senond .
	a series of display for unpacking of numbers
j)	Explain the following instructions with an example.
->	CMPXCHQ - this instructions is used to compare
	two operands & exchange

Page 2	Vo.	
Date		

BSWP- this instruction takes the contents of any 32 bit register and swap the first byte with fourth, and the second with the third PUSHA-it is used to put all the register into the Stack POPA-It is used to get words from the stack to all register.

9:2. White down the algorithm if we have to accept numbers from wers

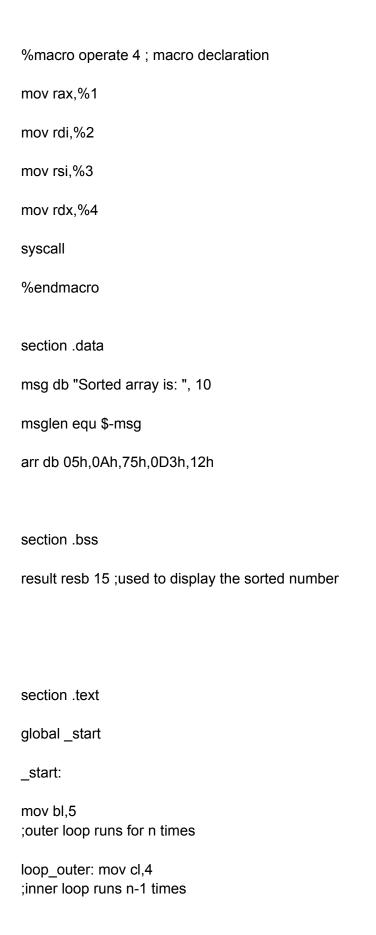
-> i) Declare variable temp

ii) Input from user the number and store it in temp

mor rax, o mor rdi, o

mov vsi, temp mov roly, 6

iv) end



```
mov rsi,arr
up: mov al,byte[rsi]
cmp al,byte[rsi+1]
jbe only_inc
;no swapping
xchg al,byte[rsi+1]
;swap
mov byte[rsi],al
only_inc: inc rsi
dec cl
;decrementing inner loop
jnz up
dec bl
;decrementing outer loop
jnz loop_outer
operate 1,1,msg, msglen
mov rdi, arr ; unpacking
mov rsi,result
mov dl,5
;for one number there are two digits
disp_loop1:
mov cl,2
```

againx: rol al,4 ;rotate by 4 mov bl,al and al,0FH cmp al,09H jbe downx ;For ascending order add al,07H downx: add al,30H mov byte[rsi],al mov al,bl inc rsi dec cl jnz againx mov byte[rsi],0AH ;inserting enter inc rsi ;result inc rdi

dec dl

mov al,[rdi]

jnz disp_loop1

operate 1,1,result,15

operate 60,0,0,0

OUTPUT

