**Exp No: 6** `**Date:**08 /10/2020

Sorting **Name:** Swetha Saseendran

**Reg No:** 185001183

## Aim:

To program and execute the sorting of 8 bit N values in ascending and descending order in 8086 microprocessor using DOSBOX.

# Programs:

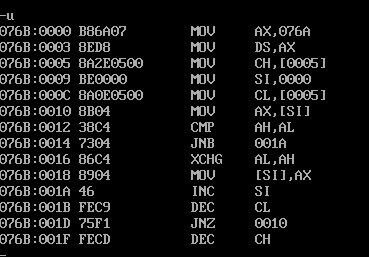
### (i) SORTING IN ASCENDING ORDER

## Algorithm:

* Program is set to run from any specified memory position.
* Load data from arr to register AX.
* Compare the digits in arr move the smaller to front and larger to back.
* Use the instruction XCHG to mov of e between the digits
* Move the digits until zero flag becomes zero and length of arr becomes zero
* Terminate the program.

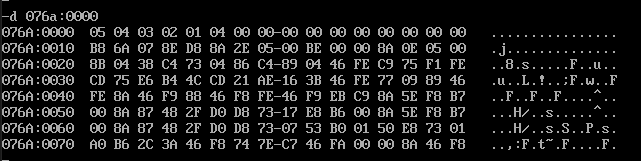
|  |  |
| --- | --- |
| **Program** | **Comments** |
| ;To sort a set of numbers in an arr in ascending order  DATA SEGMENT  arr DB 05H, 04H, 03H, 02H, 01H  arrlen DB 04H  DATA ENDS  ASSUME CS:CODE,DS:DATA  Code SEGMENT  START: MOV AX,DATA  MOV DS,AX  MOV CH, arrlen ;outer loop iteration  OUTER: MOV SI, offset(arr)  MOV CL, arrlen ;  INNER: MOV AX, [SI]  CMP AH, AL  JNC SKIP  XCHG AL, AH  MOV [SI], AX  SKIP: INC SI  DEC CL  JNZ INNER  DEC CH  JNZ OUTER  MOV AH,4CH  INT 21H  Code ENDS  END START  END | Array with 05, 04, 03, 02, 01 as input  Array length as 04  Address of data segment moved to ax  Address of ax moved to ax  Value of arrlen moved to ch  Starting pointer of arr  Inner loop iteration (reinitialize)  Jump if no carry to SKIP  AH AL is stored together  Decrease inner loop  Decease outer loop  Terminate the program |

## Unassembled Code:

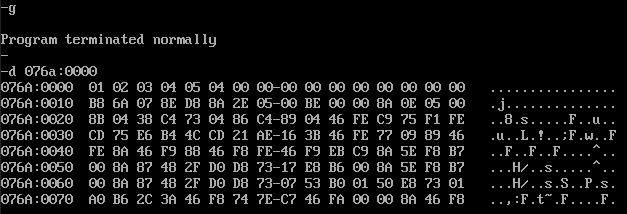
****

## Snapshot of sample input and output:

**INPUT:**



**OUTPUT:**

****

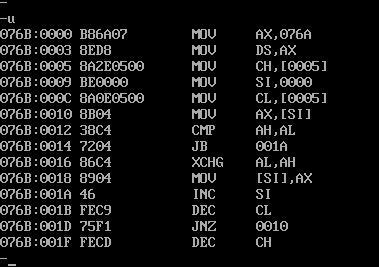
### (ii) SORTING IN DESCENDING ORDER

## Algorithm:

* Program is set to run from any specified memory position.
* Load data from arr to register AX.
* Compare the digits in arr move the larger digit to front and smaller digit to back of arr.
* Use the instruction XCHG to move between the digits
* Move the digits until zero flag becomes zero and length of arr becomes zero
* Terminate the program.

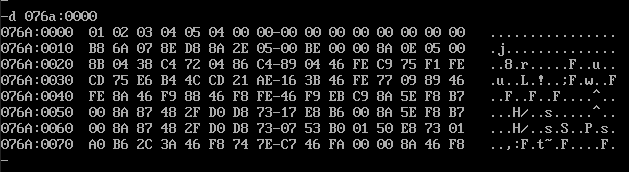
|  |  |
| --- | --- |
| **Program** | **Comments** |
| ;To sort a set of numbers in an arr in descending order  DATA SEGMENT  arr DB 01H, 02H, 03H, 04H, 05H  arrlen DB 04H  DATA ENDS  ASSUME CS:CODE,DS:DATA  Code SEGMENT  START: MOV AX,DATA  MOV DS,AX  MOV CH, arrlen  OUTER: MOV SI, offset(arr)  MOV CL, arrlen  INNER: MOV AX, [SI]  CMP AH, AL  JC SKIP  XCHG AL, AH  MOV [SI], AX  SKIP: INC SI  DEC CL  JNZ INNER  DEC CH  JNZ OUTER  MOV AH,4CH  INT 21H  Code ENDS  END START  END | Array with 01, 02, 03, 04, 05 as input  Array length as 04  Address of data segment moved to ax  Address of ax moved to ax  Value of arrlen moved to ch  Starting pointer of arr  Inner loop iteration (reinitialize)  Jump if carry to SKIP  AH AL is stored together  Decrease inner loop  Decease outer loop  Terminate the program |

## Unassembled Code:

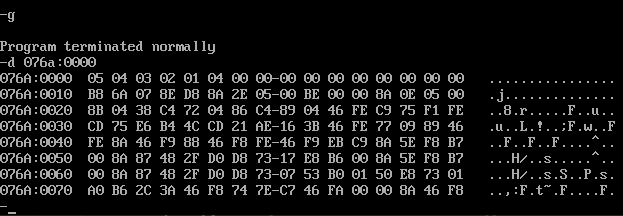


## Snapshot of sample input and output:

**INPUT:**



**OUTPUT:**



## Result:

Therefore, the ascending and descending sorting are performed and verified using MASM.