KHULNA UNIVERSITY OF ENGINEERING AND TECHNOLOGY, KUET

SESSIONAL REPORT

Course No: CSE 2204

Department of: Computer Science and Engineering

Experiment No: **8**

Name of the Experiment: To develop an assembly program that performs bubble sort

Remarks

Date of Performance: 09.05.21

Date of Submission: 19.06.21

Name: Rifat Arefin

Roll: 18070117

Year: 2nd

Semester: 2nd

No. of experiment: 08

Name of experiment: To develop an assembly program that performs bubble sort.

Objectives:

1. To learn about bubble sort algorithm.
2. To implement bubble sort algorithm with loop and other basic instructions.

Introduction:

Bubble sort algorithm is a simple sorting algorithm. This sorting algorithm is comparison-based algorithm in which each pair of adjacent elements is compared and the elements are swapped if they are not in order. This algorithm is not suitable for large

data sets as its average and worst case complexity are of $O(n^2)$ where n is the number of items.

Required Apparatus: emu 8086, laptop.

Methodology:

Code:

```
org 100h
.data
array db 3h, 4h, 1h, 6h, 8h
len equ ($-array) ; find the length of
                  array
temp db           ; temporary variable used
                  for swapping data
```

• code

```
mov cx, len     ; initialize cx with len
dec cx          ; decrement the value of cx
cmp cx, 0h      ; Compare is the value of cx is
je return           zero or not. if zero, then


outLoop:
    lea si, array   ; load the address of the
                      array to SI
    mov dx, cx      ; Store the value of cx to
                      dx for future purpose
    inLoop:
        mov al, [si] ; copy the value of si
                       memory to al register
        mov bl, [di] ; copy the value of di
                       memory to bl register
        mov di, si ; copy the address stored
                     in si to di register
        inc di      ; to point next element of
                      si
```

```
        cmp al, bl  ; comparing the values of
        ja swap       al and bl
                    ; if ja is executed then
                      jump to 'swap' label

        Back to Loop:
                inc si  ; to point next element in si
                inc di  ; to point next element in
                loop inLoop ; end of inLoop.
                          di

        mov cx, dx ; re
        loop outLoop
jmp return


swap:
        mov temp, al
        mov al, bl
        mov bl, temp
        mov [si], al
        mov [di], bl
        jmp Back to Loop
return:
        ret
```

Result and discussion:

From this experiment, we learnt how to implement bubble sort algorithm in assembly language. We tried to solve the program by using loop. We used other basic instructions. for implementing this we used jmp, cmp., loop instructions and had a cristal clear idea about the use of those instructions. We used various ~~types of~~ inputs to evaluate our program that is performed well or not. Successfully we performed well.

Conclusion:

We used bubble sort algorithm for sorting an array list and we ~~used~~ implemented this properly. We tried ~~to~~ for different inputs to ensure our program worked successfully.

References:

1. Microprocessor and Interfacing - by DV Hall

2. emu8086/ documentation /index. html.