Page No.

Date: / /

Assignment - 7

Title : Write x86 program to sort the list of integers in ascending/descending order. Read the input from the text file and write the sorted data back to the same text file using bubble sort.

Objective : Students will study sorting of number in ALP.

Outcome : Implement bubble sort in ALP.

Theory :

In bubble sort each element of the array is compared with its adjacent element. The algorithm processes the list in passes. A list with n elements requires n-1 passes for sorting. Consider an array A of n elements whose elements are to be sorted by using Bubble sort. The algorithm works in the following way.

1) ~~In~~ pass 1 A[0] is compared with A[1], A[1] is compared with A[2], A[2] is compared with A[3] and so on. At the end of pass 1, the largest element of the array is placed at the highest index of the list.

2) In Pass 2 A[0] is compared with A[1]. A[1] is compared with A[2] and so on. At the end of pass 2 the second largest element of the array is placed at the second highest index of the list.

Scanned with CamScanner

3) In pass n-1, the A[0] is compared with A[1], A[1] is compared with A[2] and so on. At the end of this pass the smallest element of the list is placed at the first index of the list.

Open File

```
    mov  rax, 2        ;  open syscall
    mov  rdi, fname    ;  fname = filename
    mov  rsi, 2        ;  file access mode
    mov  rdx, 0777     ;  Permissions set.
    syscall
```

```
            0  7  7  7
   user ───────┘  ↑  └── all
                group
```

Read File

```
  mov rax, 0          ; Read syscall
  mov rdi, [fdis]     ; File Pointer (Descriptor)
  mov rsi, buffer     ; Buffer for read
  mov rdx, length     ; len of Buffer.
```

Algorithm :

1) ~~Sort~~ Starting with the first element (index = 0), compare the current element with the next element of the array.
2) If the current element is greater than the next element of the array, swap them.
3) If the current element is less than the

next element, move to the next element. Repeat step 1.

Test cases :

	Test case	Expected	Outcome	Result
1)	4 8 2 0 3 1 9	0 1 2 3 4 8 9	As expected	Pass.
2)	9 5 3 4 5	3 4 5 5 9	As expected	Pass

Conclusion: We successfully implemented Bubble sort using ALP.