CET 214 – Data Structures & Algorithms

Experiment # 8

Experiment Title

Sorting Algorithms

Assessment of CLO(s): IV

Performed on 29-11-2024

Student Name			
Roll No.		**Group**	
Semester		**Session**	

Total (Max)	Criteria 1 (2.5)	Criteria 2 (2.5)	Criteria 3 (2.5)	Criteria 4 (2.5)	Total (10)
Marks Obtained					
Remarks (if any)					

Experiment evaluated by

Instructor's Name	**Engr. Muhammad Asad Husain**		
Date		**Signature**	

Department of Engineering Technology
(UIT University)

Course Code: CET214 Course Title: Data Structures & Algorithms Course Credits: 2+1 Session: Fall 2024

Rubric for assessment criteria to perform experiment number 7.

Level / Criteria	UNSATISFACTORY 1	COMPETENT 2	PROFICIENT 3	DISTINGUISHED 4
Capability of writing algorithm/ Procedure	None of the steps are implemented of an algorithm.	Few steps are implemented correctly of an algorithm.	Most of the steps are implemented correctly of an algorithm.	All the steps are implemented correctly of an algorithm.
Capability of writing Program	Programs not completed.	Completeness of code, consistent variable naming and unformatted.	Completeness of code, inconsistent variable naming and well formatted.	Completeness of code, consistent variable naming and well formatted.
Completion of target in Lab	25% target has been completed	50% target has been completed	75% target has been completed	100% target has been completed
Output	None of the outputs are correct.	Few outputs have been found correctly.	Some of the outputs are correct and well formatted.	Most of the outputs are correct and well formatted.

Practical Objective(s):

1. Learn how to sort an array of numbers using Bubble Sort Algorithm

Theory

Bubble Sort

Bubble sort compares each pair of adjacent items in an array and swaps them if they are in the wrong order. Let DATA be an array of N numbers. Sorting DATA refers to the operation of rearranging the elements of DATA so that they are in an increasing order i.e.

DATA[1]<DATA[2]<DATA[3]<……DATA[N]

An array with N elements will be sorted in N-1 passes. With each pass, one element will be placed in right order. Therefore, with each pass, there is one less element to be sorted. This effectively means that after the completion of each pass, a sub array needs to be sorted, which is one less than the previous pass. The sorting is completed when there is only one element left to be sorted.

Algorithm: Bubble (DATA, N)
Here DATA is an array with N elements. This element sorts the element in DATA.

Step 1. Repeat Steps 2 and 3 for PASS=1 to N-1
Step 2. Set K=0 [Initialize counter for comparison]
Step 3. Repeat while K <N-PASS

a) If DATA[K]> DATA[K+1], then
Interchange DATA [K] and DATA [K+1]
[End of If structure]

b) Set K=K+1
[End of inner loop]
[End of step 1 outer loop].

Step 4. Write: Sorted DATA
Step 5. Exit

Do It Yourself:

1. Write the code of algorithm given above.
2. Modify the program in task 1 and add a variable to count the number of passes and interchanges when size of array is 10 in the following cases:
   a) When the given array is randomly sorted.
   b) When the given array is in descending order. (Worst case)