

Assignment Description.

This Assignment requires you to implement and test the Bubble Sort and Improved Bubble Sort algorithms. Both algorithms should be implemented as functions, with a single main program to test each of them.

To test the **Bubble Sort** algorithm:

- Use an array where the elements are in reverse (descending order) from n to 1.
- Use $n = 800, 1600, 2400, 3200$ and 4000 .

To test the **Improved Bubble Sort** algorithm:

- Use an array where the elements are randomly generated using the `rand()` function, `mod (2*n)`. This reduces the chance of repeated elements in the array.
- Use $n = 800, 1600, 2400, 3200$ and 4000 .

For each of the Sorts create a Table of values containing:

n	800	1600	2400	3200	4000
No. of comparisons					
No. of swaps					

For the Bubble Sort results, use Excel to create a **line graph** plotting n (on the x-axis) and the number of comparisons done. Copy the graph into your submission document.

Assignment Submission.

Submission date: Monday 15th April, 11.00am

A **hardcopy (printed)** of the Assignment must be submitted to me (or the Lab tutor), and signature provided upon submission.

Submit also the C code only in a single .cpp file on webcourses, filename Assignment1.cpp.

Late submissions will be subject to a penalty per day. After 10 days a mark of 0 will be given. It **must** be the candidates **own** work. Students should be aware of the **General Assessment Regulations**, and in particular Appendix 1. For details see

<http://www.dit.ie/qualityassuranceandacademicprogrammerekords/student-assessment-regulations/general/>

Marking.

Correctly functioning code, with proper program constructs and good program layout/style:	(5 marks)
Tables, Graph and comments on results, with submitted document including code.	(3 marks)
Lab Demonstration.	(2 marks)