

CURTIN UNIVERSITY (CRICOS number: 00301J)
Department of Computing, Faculty of Engineering and Science
Data Structures and Algorithms (COMP1002)

PRACTICAL 10 – DSA IN PRACTICE

AIMS

- To implement and assess additional sorting algorithms
- To use Java/Python Collections to implement ADTs

BEFORE THE PRACTICAL:

- Read this practical sheet fully before starting.
- Make sure you have completed previous practicals.

ACTIVITY 1: SHELL SORT

Return to your code from the advanced sorting practical. Using the lecture notes, or other online resources, find the algorithm for Shell sort and add it to `Sorts.java` and `SortsTestHarness.java`.

Using the script **run.sh**, assess where shell sort fits compared to other sorting algorithms in terms of performance?

ACTIVITY 2: COUNTING SORT

Working with the lecture notes and online resources, add Counting Sort to `Sorts.java`. Note that you may have to change the input data to have the repetition and smaller range of values to suit this sort (you should have this from the Quicksort extension).

What are the benefits of Counting Sort, and what negatives or limitations does it have?

ACTIVITY 3: RADIX LSD SORT

Working with the lecture notes and online resources, add the Radix LSD sort to `Sorts.java`. You should use your Counting sort implementation for the inner sorting function.

Using the script **run.sh**, assess where your Radix Sort fits compared to other sorting algorithms in terms of performance? How is it impacted by choice of base?

ACTIVITY 4: JAVA / PYTHON COLLECTIONS QUEUE

Using your code from previous practicals, substitute a Java / Python collections-based stack for the `DSAQueue`. Test it using your Queue test harness. (Python: `deque`)

ACTIVITY 5: JAVA / PYTHON COLLECTIONS HASH TABLE

Using your code from the Hash Table practical, substitute a Java / Python collections-based HashMap for the DSAHashTable. Test it using your Hash Table test harness. (Python: Dictionary/Map)

SUBMISSION DELIVERABLE:

Your UML and completed java classes and related files are due before the beginning of your next tutorial.

SUBMIT ELECTRONICALLY VIA BLACKBOARD, under the *Assessments* section.

MARKING GUIDE

Your submission will be marked as follows:

- [2] Shell sort implemented and tested.
- [2] Counting sort implemented and tested.
- [2] Radix LSD sort implemented and tested.
- [2] Queue implemented through built-in functionality, and tested + UML
- [2] Hash table (map) implemented through built-in functionality, and tested + UML