5.6 Sorting

Name: _____ Class: _____ Date: _____ **Total: 24 marks**

KEY WORDS insertion sort • merge sort

Objective

Count the inversion bill, and sort by both families by hand.

1 Key knowledge and worked examples

■ The bill

The dataset $[5, 2, 9, 1, 7, 3]$ holds 8 inversions: every adjacent swap fixes exactly one, so swap-adjacent sorts pay 8 ✓. Insertion sort's shifts equal the inversions ✓.

■ Worked example: two families, one result

Insertion: shifts $1, 0, 3, 1, 3$ – total 8 shifts, 11 comparisons ✓. Merge sort on the same data: 10 comparisons – and at $n = 1000$ the families divide by about 100 ✓.

2 Practice

2.1 Count the inversions of $[3, 1, 4, 2]$. [3]

2.2 Which two of insertion, merge, quick, heap are stable? [3]

2.3 Sorted input with a first-element pivot is which sort's worst case, and of what class? [3]

3 Exam-style questions

3.1 Sort $[4, 8, 2, 5, 1]$ by insertion sort.

Give the array after each insertion and the shift count of each pass with the running total. Count the dataset's inversions and state the theorem that ties the two totals. Give the comparison count if every pass ends in one failed comparison. [8]

3.2 Sort the same $[4, 8, 2, 5, 1]$ by merge sort.

Show the halving tree down to singletons. Merge bottom-up, giving each merge's comparison count and the running total. Compare that total with insertion's on this input, and state each family's worst-case class. [7]