4.1 Decision Maths: Algorithms, Sorting and Bin Packing

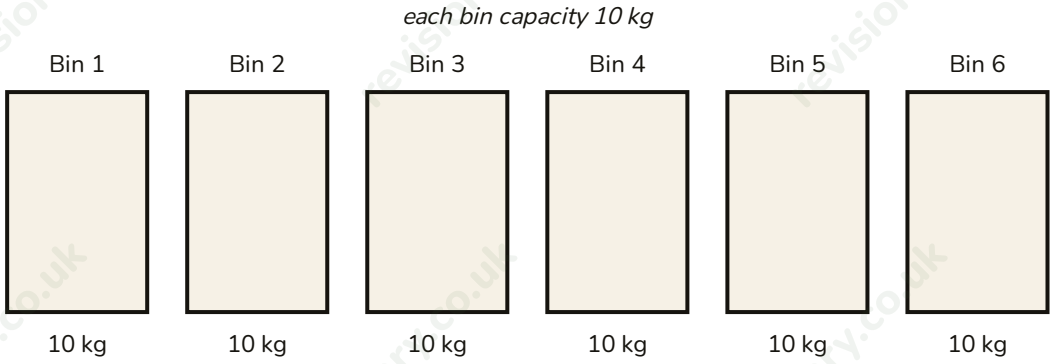
Total Marks

EDEXCEL 9FM0 · Calculator allowed · about 225 minutes · Write-in edition

Name: _____ Date: ____ / ____ / _____

Answer ALL questions. Show all your working.

1 Chidi works for a small courier company. On one shift, he must load parcels of the following weights, in kg, into delivery bins that can each hold a maximum of 10 kg:
6, 5, 4, 7, 3, 8, 2, 5



(a) Apply the first-fit algorithm to the parcels in the order given, packing them into bins of capacity 10 kg. For each bin, show which parcels it contains. State the number of bins used. **(4)**

(b) Calculate the theoretical lower bound for the number of bins needed for this set of parcels, and compare it with your answer to part (a). **(2)**

(Total for Question 1 is 6 marks)

2 This question uses known comparison-count formulas to compare the efficiency of a bubble sort and a quicksort as the size of the list, n, grows large.

(a) The number of comparisons made by a bubble sort in the worst case, on a list of n items, is given by n(n - 1)/2. Calculate the worst-case number of comparisons for (i) n = 50, (ii) n = 200, (iii) n = 1024. **(3)**

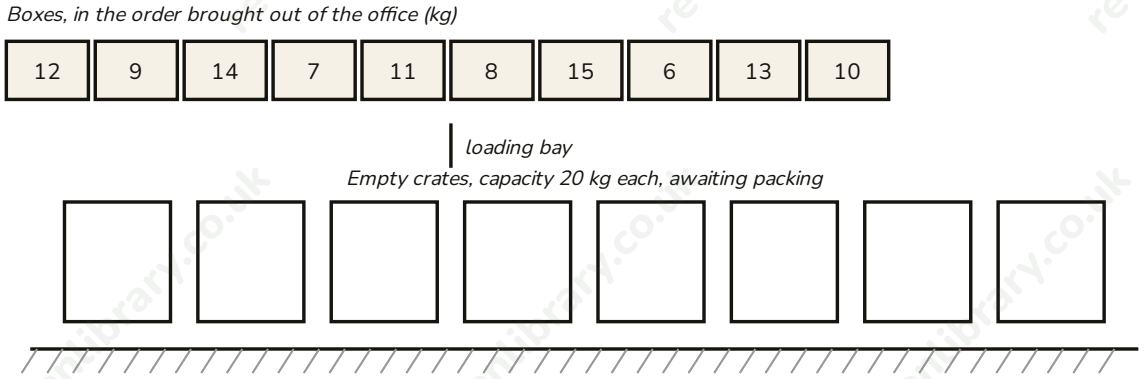
(b) Given that n increased by a factor of 4 between parts (a)(i) and (a)(ii), show that the ratio of your two answers is consistent with the bubble sort's worst-case order of $O(n^2)$. **(2)**

(c) A quicksort, on average, makes approximately n log2(n) comparisons to sort a list of n items. Calculate the approximate number of comparisons a quicksort would make, on average, to sort a list of n = 1024 items. **(2)**

(d) By comparing your answer to part (c) with your answer to part (a)(iii), comment on the practical significance, for large n, of the difference in order between an O(n log n) algorithm and an $O(n^2)$ algorithm. **(2)**

(Total for Question 2 is 9 marks)

3 Aisha runs a small office-relocation company. For one client's move, the following box weights, in kg, must be packed into crates that can each hold a maximum of 20 kg, in the order the boxes are brought out of the office:
12, 9, 14, 7, 11, 8, 15, 6, 13, 10



(a) Calculate the theoretical lower bound for the number of crates needed. **(2)**

(b) Apply first-fit to the weights in the order given, packing into crates of capacity 20 kg. Show the contents of each crate, and state the number of crates used. **(3)**

(c) Sort the weights into decreasing order and apply first-fit decreasing, packing into crates of capacity 20 kg. Show the contents of each crate, and state the number of crates used. **(3)**

(d) Using a full-bin method, find as many crates as possible that can be exactly filled from the original ten weights, and state which items (if any) cannot be paired to exactly fill a crate. **(3)**

(e) Comment on what your answers to parts (b) to (d) show about the effectiveness of first-fit, first-fit decreasing and the full-bin method for this particular set of data. **(2)**

(Total for Question 3 is 13 marks)

4 The items with the following sizes are to be packed into bins of capacity 20: 11, 15, 12, 14, 7, 11, 5, 5, 4.

(a) Find a lower bound for the number of bins needed. **(2)**

(b) Use the first-fit algorithm, taking the items in the order given, to pack the items into bins. State the contents of each bin. **(3)**

(c) Use the first-fit decreasing algorithm to pack the items. State the number of bins used and the contents of each bin. **(3)**

(Total for Question 4 is 8 marks)

5 The list 40, 1, 24, 34, 50, 55, 44, 28 is to be sorted into ascending order using a bubble sort. In each pass, adjacent items are compared from the left and swapped if they are in the wrong order.

(a) Show the list after the first pass and state the number of swaps made in the first pass. **(3)**

(b) Show the list after the second pass. **(2)**

(c) State the number of passes needed to sort the list completely, not counting a final pass in which no swaps are made, and the total number of swaps. **(2)**

(Total for Question 5 is 7 marks)

6 The list 48, 38, 40, 43, 22, 12, 49, 52 is to be sorted into ascending order using a bubble sort. In each pass, adjacent items are compared from the left and swapped if they are in the wrong order.

(a) Show the list after the first pass and state the number of swaps made in the first pass. **(3)**

(b) Show the list after the second pass. **(2)**

(c) State the number of passes needed to sort the list completely, not counting a final pass in which no swaps are made, and the total number of swaps. **(2)**

(Total for Question 6 is 7 marks)

7 The list 56, 10, 39, 2, 27, 11, 14 is to be sorted into ascending order using a bubble sort. In each pass, adjacent items are compared from the left and swapped if they are in the wrong order.

(a) Show the list after the first pass and state the number of swaps made in the first pass. **(3)**

(b) Show the list after the second pass. **(2)**

(c) State the number of passes needed to sort the list completely, not counting a final pass in which no swaps are made, and the total number of swaps. **(2)**

(Total for Question 7 is 7 marks)

8 The list 14, 2, 7, 15, 28, 51, 54, 16 is to be sorted into ascending order using a bubble sort. In each pass, adjacent items are compared from the left and swapped if they are in the wrong order.

(a) Show the list after the first pass and state the number of swaps made in the first pass. **(3)**

(b) Show the list after the second pass. **(2)**

(c) State the number of passes needed to sort the list completely, not counting a final pass in which no swaps are made, and the total number of swaps. **(2)**

(Total for Question 8 is 7 marks)

9 The list 3, 23, 46, 36, 54, 27, 43, 58, 55 is to be sorted into ascending order using a bubble sort. In each pass, adjacent items are compared from the left and swapped if they are in the wrong order.

(a) Show the list after the first pass and state the number of swaps made in the first pass. **(3)**

(b) Show the list after the second pass. **(2)**

(c) State the number of passes needed to sort the list completely, not counting a final pass in which no swaps are made, and the total number of swaps. **(2)**

(Total for Question 9 is 7 marks)

10 The list 7, 56, 49, 51, 31, 59, 28, 55, 9 is to be sorted into ascending order using a bubble sort. In each pass, adjacent items are compared from the left and swapped if they are in the wrong order.

(a) Show the list after the first pass and state the number of swaps made in the first pass. **(3)**

(b) Show the list after the second pass. **(2)**

(c) State the number of passes needed to sort the list completely, not counting a final pass in which no swaps are made, and the total number of swaps. **(2)**

(Total for Question 10 is 7 marks)

11 The items with the following sizes are to be packed into bins of capacity 40: 35, 9, 11, 12, 24, 7, 33, 35.

(a) Find a lower bound for the number of bins needed. **(2)**

(b) Use the first-fit algorithm, taking the items in the order given, to pack the items into bins. State the contents of each bin. **(3)**

(c) Use the first-fit decreasing algorithm to pack the items. State the number of bins used and the contents of each bin. **(3)**

(Total for Question 11 is 8 marks)

12 The items with the following sizes are to be packed into bins of capacity 40: 25, 21, 24, 23, 22, 29, 4, 16, 12, 36.

(a) Find a lower bound for the number of bins needed. **(2)**

(b) Use the first-fit algorithm, taking the items in the order given, to pack the items into bins. State the contents of each bin. **(3)**

(c) Use the first-fit decreasing algorithm to pack the items. State the number of bins used and the contents of each bin. **(3)**

(Total for Question 12 is 8 marks)

13 The items with the following sizes are to be packed into bins of capacity 20: 12, 14, 14, 6, 14, 7, 14, 5, 10.

(a) Find a lower bound for the number of bins needed. **(2)**

(b) Use the first-fit algorithm, taking the items in the order given, to pack the items into bins. State the contents of each bin. **(3)**

(c) Use the first-fit decreasing algorithm to pack the items. State the number of bins used and the contents of each bin. **(3)**

(Total for Question 13 is 8 marks)

14 The items with the following sizes are to be packed into bins of capacity 25: 21, 18, 14, 19, 7, 20, 19, 10, 8.

(a) Find a lower bound for the number of bins needed. **(2)**

(b) Use the first-fit algorithm, taking the items in the order given, to pack the items into bins. State the contents of each bin. **(3)**

(c) Use the first-fit decreasing algorithm to pack the items. State the number of bins used and the contents of each bin. **(3)**

(Total for Question 14 is 8 marks)

15 The list 33, 3, 14, 31, 8, 44, 56 is to be sorted into ascending order using a bubble sort. In each pass, adjacent items are compared from the left and swapped if they are in the wrong order.

(a) Show the list after the first pass and state the number of swaps made in the first pass. **(3)**

(b) Show the list after the second pass. **(2)**

(c) State the number of passes needed to sort the list completely, not counting a final pass in which no swaps are made, and the total number of swaps. **(2)**

(Total for Question 15 is 7 marks)

16 The items with the following sizes are to be packed into bins of capacity 50: 8, 11, 34, 39, 35, 31, 18, 13.

(a) Find a lower bound for the number of bins needed. **(2)**

(b) Use the first-fit algorithm, taking the items in the order given, to pack the items into bins. State the contents of each bin. **(3)**

(c) Use the first-fit decreasing algorithm to pack the items. State the number of bins used and the contents of each bin. **(3)**

(Total for Question 16 is 8 marks)

17 The items with the following sizes are to be packed into bins of capacity 20: 8, 8, 8, 14, 15, 10, 16, 9, 9, 5.

(a) Find a lower bound for the number of bins needed. **(2)**

(b) Use the first-fit algorithm, taking the items in the order given, to pack the items into bins. State the contents of each bin. **(3)**

(c) Use the first-fit decreasing algorithm to pack the items. State the number of bins used and the contents of each bin. **(3)**

(Total for Question 17 is 8 marks)

18 The items with the following sizes are to be packed into bins of capacity 20: 14, 4, 7, 5, 8, 3, 6, 3.

(a) Find a lower bound for the number of bins needed. **(2)**

(b) Use the first-fit algorithm, taking the items in the order given, to pack the items into bins. State the contents of each bin. **(3)**

(c) Use the first-fit decreasing algorithm to pack the items. State the number of bins used and the contents of each bin. **(3)**

(Total for Question 18 is 8 marks)

19 The list 26, 57, 14, 34, 16, 48, 3 is to be sorted into ascending order using a bubble sort. In each pass, adjacent items are compared from the left and swapped if they are in the wrong order.

(a) Show the list after the first pass and state the number of swaps made in the first pass. **(3)**

(b) Show the list after the second pass. **(2)**

(c) State the number of passes needed to sort the list completely, not counting a final pass in which no swaps are made, and the total number of swaps. **(2)**

(Total for Question 19 is 7 marks)

20 A courier drives to eight different addresses in one shift. The distances, in miles, driven to reach each address, in the order the deliveries were planned, are:
34, 12, 55, 8, 41, 27, 63, 19

(a) Using the first item of each sublist as the pivot, perform a quicksort on this list to sort it into ascending order. At each stage, state the pivot used and the two resulting sublists (items less than the pivot, and items greater than or equal to the pivot). Continue until every sublist contains at most one item, then write out the final sorted list. **(5)**

(b) State the total number of comparisons used to complete the sort, given that each non-pivot item in a sublist is compared exactly once with that sublist's pivot. **(2)**

(c) State one advantage and one disadvantage of always choosing the first item of a sublist as the pivot, compared with other pivot-selection methods. **(2)**

(Total for Question 20 is 9 marks)

---

21 This question continues from question 1, using the same eight parcel weights and the same 10 kg bin capacity.

(a) Re-order Chidi's parcel weights into decreasing order of size, then apply the first-fit algorithm to this re-ordered list (this is called first-fit decreasing). Show the contents of each bin, and state the number of bins used. **(3)**

(b) State whether first-fit decreasing has achieved the lower bound found in question 1(b), and explain why sorting the weights into decreasing order before applying first-fit can reduce the number of bins needed, compared with using first-fit on the original order. **(2)**

(Total for Question 21 is 5 marks)

22 Freya, an art technician at a secondary school, has eight tins of paint with the following volumes, in litres, to store in crates that can each hold a maximum of 10 litres:
8, 2, 7, 3, 6, 4, 5, 5

(a) Calculate the theoretical lower bound for the number of crates needed to hold all eight tins. **(2)**

(b) Use a full-bin method: starting with the largest unallocated tin each time, try to find one or more other tins whose volumes sum to exactly fill the remaining space in that crate. Show the contents of each crate you form. **(4)**

(c) Compare your answer to part (b) with your lower bound in part (a), and state what this shows about this particular set of tin volumes. **(1)**

(Total for Question 22 is 7 marks)

23 This question is about determining the order of an algorithm from timed test results.

(a) Ravi times how long Algorithm 1 takes to run on lists of different sizes, n. His results are shown below. **(3)**

n = 10, time = 8 ms
n = 20, time = 32 ms
n = 40, time = 128 ms
n = 80, time = 512 ms

Use a doubling ratio test (comparing the run time each time n doubles) to estimate the order of Algorithm 1, giving your answer in the form $O(n^k)$.

n	Time (ms)
10	8
20	32
40	128
80	512

(b) Ravi also tests Algorithm 2, obtaining the results below. **(2)**

n = 5, time = 4 ms
n = 10, time = 32 ms
n = 20, time = 256 ms

Use the same method to estimate the order of Algorithm 2.

n	Time (ms)
5	4
10	32
20	256

(c) A third algorithm has order O(n). It takes 15 ms to process a list of 200 items. Estimate how long it would take this algorithm to process a list of 800 items, showing your reasoning. **(2)**

(Total for Question 23 is 7 marks)

Mark scheme · 4.1 Decision Maths: Algorithms, Sorting and Bin Packing

Question 1

(a) correct method: each parcel placed in the first bin (in order) with enough remaining space, opening a new bin only when none of the existing bins can fit it **M1**

(a) correct bin contents: Bin 1 {6, 4}, Bin 2 {5, 3, 2}, Bin 3 {7}, Bin 4 {8}, Bin 5 {5} **A1**

(a) 5 bins used, cao **B1**

(a) correctly states the spare capacity in each bin: 0 kg, 0 kg, 3 kg, 2 kg, 5 kg (total 10 kg wasted) **B1**

(a) Answer: Bin 1: 6, 4 (full). Bin 2: 5, 3, 2 (full). Bin 3: 7 (3 kg spare). Bin 4: 8 (2 kg spare). Bin 5: 5 (5 kg spare). 5 bins used.

(b) total weight = 6+5+4+7+3+8+2+5 = 40 kg, lower bound = ceiling(40/10) **M1**

(b) lower bound = 4 bins, and correctly notes that first-fit's 5 bins is one more than this lower bound **A1**

(b) Answer: Lower bound = 4 bins. First-fit used 5 bins, one more than this lower bound.

Question 2

(a) (i) 1225 cao (50 x 49 / 2) **B1**

(a) (ii) 19900 cao (200 x 199 / 2) **B1**

(a) (iii) 523776 cao (1024 x 1023 / 2) **B1**

(a) Answer: (i) 1225; (ii) 19900; (iii) 523776.

(b) correctly calculates the ratio 19900 / 1225 = 16.24... (awrt 16.2) **M1**

(b) compares this to $4^2 = 16$ and concludes the closeness confirms the order $O(n^2)$ (the small difference from exactly 16 is because n(n-1)/2 also has a lower-order term, becoming negligible for large n) (oe) **A1**

(b) Answer: Ratio approx 16.2, close to $4^2 = 16$, consistent with worst-case order $O(n^2)$.

(c) uses log2(1024) = 10 (since $2^{10} = 1024$) **M1**

(c) 1024 x 10 = 10240 comparisons cao **A1**

(c) Answer: Approximately 10240 comparisons.

(d) for n = 1024, quicksort's average 10240 comparisons is far smaller than bubble sort's worst-case 523776 comparisons, roughly 51 times fewer (oe, allow awrt 50) **B1**

(d) as n grows large, an O(n log n) algorithm remains practical while an $O(n^2)$ algorithm's run time grows far more quickly, so the difference in order becomes increasingly significant for large data sets (oe) **B1**

(d) Answer: Quicksort's estimate (10240) is roughly 51 times smaller than bubble sort's worst case (523776); as n grows, this gap widens further, so the choice of algorithm order matters far more for large data sets.

Question 3

(a) total weight = 12+9+14+7+11+8+15+6+13+10 = 105 kg **M1**

(a) lower bound = ceiling(105/20) = 6 cao **A1**

(a) Answer: Lower bound = 6 crates.

(b) correct method applied in the given order, checking each weight against existing crates before opening a new one **M1**

(b) correct crates: {12,7}, {9,11}, {14,6}, {8,10}, {15}, {13} **A1**

(b) 6 crates used cao **B1**

(b) Answer: Crates: {12,7}, {9,11}, {14,6}, {8,10}, {15}, {13}. 6 crates used.

(c) correct decreasing order: 15, 14, 13, 12, 11, 10, 9, 8, 7, 6 **M1**

(c) correct crates: {15}, {14,6}, {13,7}, {12,8}, {11,9}, {10} **A1**

(c) 6 crates used cao **B1**

(c) Answer: Crates: {15}, {14,6}, {13,7}, {12,8}, {11,9}, {10}. 6 crates used.

(d) identifies the four pairs that sum to exactly 20: (12,8), (9,11), (14,6), (7,13) **M1**

(d) correctly identifies that the two remaining items, 15 and 10, cannot be combined (15+10 = 25 > 20), so each needs its own crate **A1**

(d) total crates still 6 cao (4 full crates plus 2 further crates for 15 and 10), matching the lower bound in part (a) **B1**

(d) Answer: Full crates: {12,8}, {9,11}, {14,6}, {7,13}. Remaining items 15 and 10 cannot be paired (sum 25 > 20), so each needs its own crate. Total 6 crates.

(e) all three methods achieve the same number of crates, 6, matching the lower bound from part (a), so an optimal packing has been found for this data set (oe) **B1**

(e) this agreement is not guaranteed in general: as seen in earlier questions, first-fit alone does not always reach the lower bound, so its success here depends on the particular set of weights rather than being a general property of the algorithm (oe) **B1**

(e) Answer: All three methods reach the optimal 6 crates for this data set, but this is a feature of this particular set of weights rather than a guarantee: first-fit does not always match the lower bound (as seen with Chidi's parcels in question 1).

Question 4

(a) total size = 84 **M1**

(a) 84/20 = 4.20, so at least 5 bins **A1**

(a) Answer: 5

(b) places each item in the first bin with enough room, in the order given **M1**

(b) first bins correct **A1**

(b) bins: (11, 7); (15, 5); (12, 5); (14, 4); (11) **A1**

(b) Answer: (11, 7); (15, 5); (12, 5); (14, 4); (11)

(c) sorts the items into descending order **M1**

(c) applies first-fit to the sorted list **M1**

(c) 5 bins: (15, 5); (14, 5); (12, 7); (11, 4); (11) **A1**

(c) Answer: 5 bins: (15, 5); (14, 5); (12, 7); (11, 4); (11)

Question 5

(a) compares adjacent pairs from the left, swapping when the left item is larger **M1**

(a) list after pass 1: 1, 24, 34, 40, 50, 44, 28, 55 **A1**

(a) 5 swaps **B1**

(a) Answer: 1, 24, 34, 40, 50, 44, 28, 55; 5 swaps

(b) repeats the process, ignoring the final item (already in place) **M1**

(b) 1, 24, 34, 40, 44, 28, 50, 55 **A1**

(b) Answer: 1, 24, 34, 40, 44, 28, 50, 55

(c) 5 passes **B1**

(c) 10 swaps in total **B1**

(c) Answer: 5 passes, 10 swaps

Question 6

(a) compares adjacent pairs from the left, swapping when the left item is larger **M1**

(a) list after pass 1: 38, 40, 43, 22, 12, 48, 49, 52 **A1**

(a) 5 swaps **B1**

(a) Answer: 38, 40, 43, 22, 12, 48, 49, 52; 5 swaps

(b) repeats the process, ignoring the final item (already in place) **M1**

(b) 38, 40, 22, 12, 43, 48, 49, 52 **A1**

(b) Answer: 38, 40, 22, 12, 43, 48, 49, 52

(c) 5 passes **B1**

(c) 12 swaps in total **B1**

(c) Answer: 5 passes, 12 swaps

Question 7

(a) compares adjacent pairs from the left, swapping when the left item is larger **M1**

(a) list after pass 1: 10, 39, 2, 27, 11, 14, 56 **A1**

(a) 6 swaps **B1**

(a) Answer: 10, 39, 2, 27, 11, 14, 56; 6 swaps

(b) repeats the process, ignoring the final item (already in place) **M1**

(b) 10, 2, 27, 11, 14, 39, 56 **A1**

(b) Answer: 10, 2, 27, 11, 14, 39, 56

(c) 3 passes **B1**

(c) 13 swaps in total **B1**

(c) Answer: 3 passes, 13 swaps

Question 8

(a) compares adjacent pairs from the left, swapping when the left item is larger **M1**

(a) list after pass 1: 2, 7, 14, 15, 28, 51, 16, 54 **A1**

(a) 3 swaps **B1**

(a) Answer: 2, 7, 14, 15, 28, 51, 16, 54; 3 swaps

(b) repeats the process, ignoring the final item (already in place) **M1**

(b) 2, 7, 14, 15, 28, 16, 51, 54 **A1**

(b) Answer: 2, 7, 14, 15, 28, 16, 51, 54

(c) 3 passes **B1**

(c) 5 swaps in total **B1**

(c) Answer: 3 passes, 5 swaps

Question 9

(a) compares adjacent pairs from the left, swapping when the left item is larger **M1**

(a) list after pass 1: 3, 23, 36, 46, 27, 43, 54, 55, 58 **A1**

(a) 4 swaps **B1**

(a) Answer: 3, 23, 36, 46, 27, 43, 54, 55, 58; 4 swaps

(b) repeats the process, ignoring the final item (already in place) **M1**

(b) 3, 23, 36, 27, 43, 46, 54, 55, 58 **A1**

(b) Answer: 3, 23, 36, 27, 43, 46, 54, 55, 58

(c) 3 passes **B1**

(c) 7 swaps in total **B1**

(c) Answer: 3 passes, 7 swaps

Question 10

(a) compares adjacent pairs from the left, swapping when the left item is larger **M1**

(a) list after pass 1: 7, 49, 51, 31, 56, 28, 55, 9, 59 **A1**

(a) 6 swaps **B1**

(a) Answer: 7, 49, 51, 31, 56, 28, 55, 9, 59; 6 swaps

(b) repeats the process, ignoring the final item (already in place) **M1**

(b) 7, 49, 31, 51, 28, 55, 9, 56, 59 **A1**

(b) Answer: 7, 49, 31, 51, 28, 55, 9, 56, 59

(c) 7 passes **B1**

(c) 19 swaps in total **B1**

(c) Answer: 7 passes, 19 swaps

Question 11

(a) total size = 166 **M1**

(a) 166/40 = 4.15, so at least 5 bins **A1**

(a) Answer: 5

(b) places each item in the first bin with enough room, in the order given **M1**

(b) first bins correct **A1**

(b) bins: (35); (9, 11, 12, 7); (24); (33); (35) **A1**

(b) Answer: (35); (9, 11, 12, 7); (24); (33); (35)

(c) sorts the items into descending order **M1**

(c) applies first-fit to the sorted list **M1**

(c) 5 bins: (35); (35); (33, 7); (24, 12); (11, 9) **A1**

(c) Answer: 5 bins: (35); (35); (33, 7); (24, 12); (11, 9)

Question 12

(a) total size = 212 **M1**

(a) 212/40 = 5.30, so at least 6 bins **A1**

(a) Answer: 6

(b) places each item in the first bin with enough room, in the order given **M1**

(b) first bins correct **A1**

(b) bins: (25, 4); (21, 16); (24, 12); (23); (22); (29); (36) **A1**

(b) Answer: (25, 4); (21, 16); (24, 12); (23); (22); (29); (36)

(c) sorts the items into descending order **M1**

(c) applies first-fit to the sorted list **M1**

(c) 7 bins: (36, 4); (29); (25, 12); (24, 16); (23); (22); (21) **A1**

(c) Answer: 7 bins: (36, 4); (29); (25, 12); (24, 16); (23); (22); (21)

Question 13

(a) total size = 96 **M1**

(a) 96/20 = 4.80, so at least 5 bins **A1**

(a) Answer: 5

(b) places each item in the first bin with enough room, in the order given **M1**

(b) first bins correct **A1**

(b) bins: (12, 6); (14, 5); (14); (14); (7, 10); (14) **A1**

(b) Answer: (12, 6); (14, 5); (14); (14); (7, 10); (14)

(c) sorts the items into descending order **M1**

(c) applies first-fit to the sorted list **M1**

(c) 6 bins: (14, 6); (14, 5); (14); (14); (12, 7); (10) **A1**

(c) Answer: 6 bins: (14, 6); (14, 5); (14); (14); (12, 7); (10)

Question 14

(a) total size = 136 **M1**

(a) 136/25 = 5.44, so at least 6 bins **A1**

(a) Answer: 6

(b) places each item in the first bin with enough room, in the order given **M1**

(b) first bins correct **A1**

(b) bins: (21); (18, 7); (14, 10); (19); (20); (19); (8) **A1**

(b) Answer: (21); (18, 7); (14, 10); (19); (20); (19); (8)

(c) sorts the items into descending order **M1**

(c) applies first-fit to the sorted list **M1**

(c) 7 bins: (21); (20); (19); (19); (18, 7); (14, 10); (8) **A1**

(c) Answer: 7 bins: (21); (20); (19); (19); (18, 7); (14, 10); (8)

Question 15

(a) compares adjacent pairs from the left, swapping when the left item is larger **M1**

(a) list after pass 1: 3, 14, 31, 8, 33, 44, 56 **A1**

(a) 4 swaps **B1**

(a) Answer: 3, 14, 31, 8, 33, 44, 56; 4 swaps

(b) repeats the process, ignoring the final item (already in place) **M1**

(b) 3, 14, 8, 31, 33, 44, 56 **A1**

(b) Answer: 3, 14, 8, 31, 33, 44, 56

(c) 3 passes **B1**

(c) 6 swaps in total **B1**

(c) Answer: 3 passes, 6 swaps

Question 16

(a) total size = 189 **M1**

(a) 189/50 = 3.78, so at least 4 bins **A1**

(a) Answer: 4

(b) places each item in the first bin with enough room, in the order given **M1**

(b) first bins correct **A1**

(b) bins: (8, 11, 31); (34, 13); (39); (35); (18) **A1**

(b) Answer: (8, 11, 31); (34, 13); (39); (35); (18)

(c) sorts the items into descending order **M1**

(c) applies first-fit to the sorted list **M1**

(c) 4 bins: (39, 11); (35, 13); (34, 8); (31, 18) **A1**

(c) Answer: 4 bins: (39, 11); (35, 13); (34, 8); (31, 18)

Question 17

(a) total size = 102 **M1**

(a) 102/20 = 5.10, so at least 6 bins **A1**

(a) Answer: 6

(b) places each item in the first bin with enough room, in the order given **M1**

(b) first bins correct **A1**

(b) bins: (8, 8); (8, 10); (14, 5); (15); (16); (9, 9) **A1**

(b) Answer: (8, 8); (8, 10); (14, 5); (15); (16); (9, 9)

(c) sorts the items into descending order **M1**

(c) applies first-fit to the sorted list **M1**

(c) 6 bins: (16); (15, 5); (14); (10, 9); (9, 8); (8, 8) **A1**

(c) Answer: 6 bins: (16); (15, 5); (14); (10, 9); (9, 8); (8, 8)

Question 18

(a) total size = 50 **M1**

(a) 50/20 = 2.50, so at least 3 bins **A1**

(a) Answer: 3

(b) places each item in the first bin with enough room, in the order given **M1**

(b) first bins correct **A1**

(b) bins: (14, 4); (7, 5, 8); (3, 6, 3) **A1**

(b) Answer: (14, 4); (7, 5, 8); (3, 6, 3)

(c) sorts the items into descending order **M1**

(c) applies first-fit to the sorted list **M1**

(c) 3 bins: (14, 6); (8, 7, 5); (4, 3, 3) **A1**

(c) Answer: 3 bins: (14, 6); (8, 7, 5); (4, 3, 3)

Question 19

(a) compares adjacent pairs from the left, swapping when the left item is larger **M1**

(a) list after pass 1: 26, 14, 34, 16, 48, 3, 57 **A1**

(a) 5 swaps **B1**

(a) Answer: 26, 14, 34, 16, 48, 3, 57; 5 swaps

(b) repeats the process, ignoring the final item (already in place) **M1**

(b) 14, 26, 16, 34, 3, 48, 57 **A1**

(b) Answer: 14, 26, 16, 34, 3, 48, 57

(c) 6 passes **B1**

(c) 13 swaps in total **B1**

(c) Answer: 6 passes, 13 swaps

Question 20

(a) first pivot = 34 identified, with correct sublists {12, 8, 27, 19} (< 34) and {55, 41, 63} (≥ 34) **M1**

(a) correct second pivot and split on {12, 8, 27, 19}: pivot 12, giving {8} and {27, 19} **A1**

(a) correct second pivot and split on {55, 41, 63}: pivot 55, giving {41} and {63} **A1**

(a) correct third-level split on {27, 19}: pivot 27, giving {19} and { } (empty) **A1**

(a) correct final sorted list 8, 12, 19, 27, 34, 41, 55, 63 cao **A1**

(a) Answer: Final sorted list: 8, 12, 19, 27, 34, 41, 55, 63.

(b) correct method: sums the number of non-pivot items compared at each level, 7 + 3 + 2 + 1 **M1**

(b) 13 comparisons cao **A1**

(b) Answer: 13 comparisons.

(c) advantage: it is quick and simple to apply, since no extra comparisons are needed to choose the pivot (oe) **B1**

(c) disadvantage: if a sublist happens to be already sorted (or reverse sorted), this choice gives very unbalanced splits, leading to the algorithm's worst-case order of $O(n^2)$ (oe) **B1**

(c) Answer: Advantage: quick to select, no extra comparisons needed. Disadvantage: gives poor (unbalanced) splits, and hence poor efficiency, on lists that are already sorted or nearly sorted.

Question 21

(a) correct decreasing order: 8, 7, 6, 5, 5, 4, 3, 2 **M1**

(a) correct bins: Bin 1 {8, 2}, Bin 2 {7, 3}, Bin 3 {6, 4}, Bin 4 {5, 5}, all exactly full **A1**

(a) 4 bins used, cao **B1**

(a) Answer: Decreasing order: 8, 7, 6, 5, 5, 4, 3, 2. Bins: {8,2}, {7,3}, {6,4}, {5,5}. 4 bins used.

(b) yes, 4 bins matches the lower bound of 4 found in question 1(b), so this packing is optimal **B1**

(b) placing the largest items first means each bin's remaining space is used to accommodate progressively smaller items that fit neatly into the gaps left, avoiding the situation (seen in question 1) where small items fill early bins and leave awkward gaps too small for the larger items that come later (oe) **B1**

(b) Answer: Yes, first-fit decreasing achieves the lower bound of 4 bins. Placing the largest parcels first avoids leaving awkward small gaps that later, larger parcels cannot use.

Question 22

(a) total volume = 8+2+7+3+6+4+5+5 = 40 litres **M1**

(a) lower bound = ceiling(40/10) = 4 cao **A1**

(a) Answer: Lower bound = 4 crates.

(b) correctly pairs the largest remaining tin, 8, with the 2-litre tin to exactly fill a crate: {8, 2} **M1**

(b) correctly pairs the next largest remaining tin, 7, with the 3-litre tin: {7, 3} **A1**

(b) correctly pairs the next largest remaining tin, 6, with the 4-litre tin: {6, 4} **A1**

(b) correctly places the two remaining 5-litre tins together to exactly fill the last crate: {5, 5} **A1**

(b) Answer: Crates: {8,2}, {7,3}, {6,4}, {5,5}, each exactly full (10 litres).

(c) all 4 crates are exactly full with no wasted space, so the lower bound of 4 found in part (a) has been achieved: this packing is optimal (oe) **B1**

(c) Answer: The full-bin method achieves exactly 4 crates, matching the lower bound, so this is an optimal packing with no wasted capacity.

Question 23

(a) correctly calculates the ratio of successive run times: 32/8 = 4, 128/32 = 4, 512/128 = 4 **M1**

(a) identifies that the ratio is consistently 4 each time n doubles **A1**

(a) correctly concludes the order is $O(n^2)$, since $4 = 2^2$ oe **A1**

(a) Answer: Algorithm 1 has order $O(n^2)$.

(b) correctly calculates the ratio of successive run times: 32/4 = 8, 256/32 = 8 **M1**

(b) correctly concludes the order is $O(n^3)$, since $8 = 2^3$ oe **A1**

(b) Answer: Algorithm 2 has order $O(n^3)$.

(c) recognises that 800 is 4 times 200, and that for an O(n) algorithm run time scales by the same factor as n **M1**

(c) 60 ms cao **A1**

(c) Answer: 60 ms.