NANYANG TECHNOLOGICAL UNIVERSITY

SEMESTER 1 EXAMINATION 2021-2022

CE2101/CZ2101 – ALGORITHM DESIGN AND ANALYSIS

Nov/Dec 2021 Time Allowed: 2 hours

INSTRUCTIONS

1. This paper contains 4 questions and comprises 10 pages.

2. Answer **ALL** questions.

3. This is a closed-book examination.

4. Questions do NOT carry equal marks.

5. Use the **white answer book** to give your answers and the space at the end of the examination paper for rough work.

---

1. Based on your solution, select the capital letter of the answer to each of the following questions. Then in the answer book, shade the circle in the column of the capital letter.

   (a) A weak point of the InsertionSort algorithm is

   A. Perform poorly when the unordered list is almost sorted.
   B. When starting, some entries may not be in the final position.
   C. For a new insertion, movements for inserted entries in ordered list are always performed.
   D. When an entry is inserted, it may still not be in the final position.
   E. For a new insertion, movements for inserted entries in ordered list are not performed.
   F. When completed, some entries may still not be in the final position.

   (5 marks)

Note: Question 1 continues on Page 2

(b) Given n entries, the time complexity of the MergeSort algorithm is

A. Always $O(n \log n)$.
B. Always $O(n^2)$.
C. $O(n \log n)$ only for the Best Case.
D. $O(n^2)$ only for the Best Case.
E. $O(n \log n)$ only for the Worst Case.
F. $O(n^2)$ only for the Worst Case.

(5 marks)

(c) Which statement on the QuickSort algorithm is NOT true?

A. Always selection of one element in array as pivot.
B. Always partitioning of the list into two sublists with respect to the pivot.
C. Always partitioning of the list into two sublists with respect to the median element.
D. Recursive partitioning until the input list has one or zero element.
E. The pivot found during partitioning is already at its final position.
F. All elements in the right sublist are greater than or equal to the pivot.

(5 marks)

(d) Given n entries, which statement on the time complexity of the HeapSort algorithm is true?

A. Only the Best case is $O(n \log n)$.
B. Only the Worst case is $O(n^2)$.
C. The Best case is $O(n \log n)$ and the Worst case is $O(n^2)$.
D. The Best case is $O(n)$ and the Worst case is $O(n \log n)$.
E. Both the Best case and the Worst case are $O(n^2)$.
F. Both the Best case and Worst case are $O(n \log n)$.

(5 marks)

Note: Question 1 continues on Page 3

(e) The Dijkstra's algorithm

A. Is to find the possible paths from a single source vertex to all other vertices.
B. Is to find the possible paths from all other vertices to a single destination vertex.
C. Is to find the possible paths from all source vertices to all other vertices.
D. Is to find the shortest paths from a single source vertex to a single destination vertex.
E. Is to find the shortest paths from a single source vertex to all other vertices.
F. Is to find the shortest paths from all other vertices to a single destination vertex.

(5 marks)

(f) Which statement on the Kruskal's algorithm is NOT true?

A. It only applies to graphs in which all the links are undirected.
B. It starts off with partitioning the vertices in a graph into |E| equivalence classes, where |E| is the number of edges.
C. It processes the edges in increasing order of weight.
D. When an edge is added, the two equivalence classes that contain the nodes of the edge are combined.
E. It is a greedy algorithm.
F. It is to find a Minimum Spanning Tree.

(5 marks)

2. The adjacency matrix representation of an undirected graph is in Figure Q2, in which the element in (row, column) denotes the edge weight. For examples, there is an edge between node A and node B, denoted by (A, B) with a weight of 2; and there is no edge between node A and node E, denoted by (A, E) with a weight of ∞.

Note: Question 2 continues on Page 4

	A	B	C	D	E	G
A	0	2	∞	∞	∞	3
B	2	0	4	∞	∞	6
C	∞	4	0	∞	2	∞
D	∞	∞	∞	0	2	1
E	∞	∞	2	2	0	3
G	3	6	∞	1	3	0

Figure Q2

Starting off with node G, perform the Prim's algorithm with this graph for a Minimum Spanning Tree (MST) in iterations. Show the followings in the answer book:

(a) A solid line of each edge with a number of its weight in the **initial graph**.

(2 marks)

(b) A solid line of each edge in the partial MST and a dashed line connecting a fringe vertex to a tree vertex **after the 1st iteration**.

(3 marks)

(c) A solid line of each edge in the partial MST and a dashed line connecting a fringe vertex to a tree vertex **after the 2nd iteration**.

(3 marks)

(d) A solid line of each edge in the partial MST and a dashed line connecting a fringe vertex to a tree vertex **after the 3rd iteration**.

(3 marks)

(e) A solid line of each edge in the partial MST and a dashed line connecting a fringe vertex to a tree vertex **after the 4th iteration**.

(3 marks)

(f) A solid line of each edge in the partial MST and a dashed line connecting a fringe vertex to a tree vertex **after the 5th iteration**.

(3 marks)

Note: Question 2 continues on Page 5

(g) A solid line of each edge in the MST and a dashed line connecting a fringe vertex to a tree vertex **after the 6th iteration**.

(3 marks)

3. (a) Solve the following recurrence by the master method.

$$W(n) = 3W(n/3) + n - 1$$

(4 marks)

(b) Solve the following recurrence together with the initial conditions given by finding its characteristic root(s).

$$a_n = 9a_{n-2} \text{ for all } n \geq 2, \; a_0 = 1, a_1 = 3$$

(6 marks)

(c) (i) The Boyer-Moore algorithm performs preprocessing to compute the two arrays charJump and matchJump. Find them for the string pattern "BAAAA".

(5 marks)

(ii) When the simple Boyer-Moore algorithm and the Boyer-Moore algorithm are applied respectively to search for the pattern "BAAAA" in the text that has 20 'A's, how many character comparisons are done in each case? You only need to give the total number of comparisons but you may give your reasoning in a couple of lines as well.

(10 marks)

4. (a) *S1* is a sequence of *n1* characters and *S2* is a sequence of *n2* characters. All characters are from the set {'a', 'c', 'g', 't'}. An alignment is defined by inserting any number of character '_' (the underscore character) into *S1* and *S2* so that the resulting sequences *S1'*and *S2'* are of equal length. Each character in *S1'* has to be aligned with the same character or an underscore in the same position in *S2'* and vice versa. The cost of an alignment of *S1* and *S2* is defined as the number of underscore characters inserted in *S1* and *S2*. For example, *S1* = "ctatg" and *S2* = "ttaagc". One possible alignment is

Note: Question 4 continues on Page 6

S1' = "ct_at_g_" and
S2' = "_tta_agc"

Both *S1'* and *S2'* have length 8 and the cost is 5. We want to find the minimum cost of aligning two sequences, denoted as alignment(*n1*, *n2*).

(i) Give a recursive definition of alignment(*n1*, *n2*).

(6 marks)

(ii) Draw the subproblem graph for alignment(3, 4).

(5 marks)

(iii) Design a dynamic programming algorithm of alignment(*n1*, *n2*) using the bottom-up approach.

(6 marks)

(b) We have a knapsack of size 20 and 5 objects. The profits and the sizes of the objects are given in Table Q4. Find a subset of the objects that fits in the knapsack that maximizes the total profit by the greedy heuristic algorithm. What is the time complexity of the greedy heuristic algorithm?

Table Q4

P (profit)	10	40	35	45	6
S (size)	5	4	7	5	3

(8 marks)

END OF PAPER

ROUGH WORK

ROUGH WORK

ROUGH WORK

ROUGH WORK

CE2101 ALGORITHM DESIGN & ANALYSIS
CZ2101 ALGORITHM DESIGN & ANALYSIS

Please read the following instructions carefully:

1. **Please do not turn over the question paper until you are told to do so. Disciplinary action may be taken against you if you do so.**

2. You are not allowed to leave the examination hall unless accompanied by an invigilator. You may raise your hand if you need to communicate with the invigilator.

3. Please write your Matriculation Number on the front of the answer book.

4. Please indicate clearly in the answer book (at the appropriate place) if you are continuing the answer to a question elsewhere in the book.