

## Design and Analysis of Algorithms (CS2009)

### Course Instructor(s):

Ms. Amina Siddique, Mr. Arslan Aslam

Section(s): CY (A,B, C,D)

## Sessional-II Exam

Total Time (Hrs): 1

Total Marks: 70

Total Questions: 2

Date: Nov 4, 2024

Roll No

Course Section

Student Signature

Do not write below this line.

Attempt all the questions.

	Q1	Q2-a	Q2-b	Total
Marks	50	6	14	70
Obtained Marks				

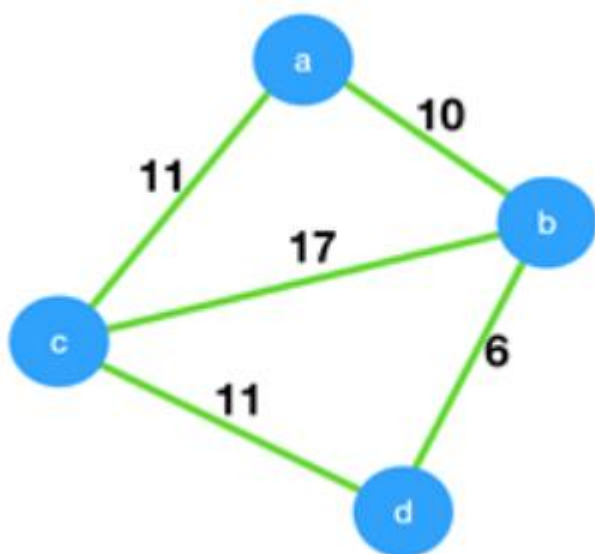
### Instructions for Multiple Choice Questions (MCQs)

- Answer all MCQs on the provided answer sheet at the end of the exam.
- Use a dark pencil or pen to fill in the bubbles completely.
- No marks will be awarded for responses not filled in on the answer sheet.
- Make sure to double-check your answers before submission, as incomplete or ambiguous responses may result in no credit for those questions.

**Q1:** Multiple Choice Questions.

**[50 marks]**

1. Consider the following graph. What is the weight of the minimum spanning tree using the Prim's algorithm, starting from vertex a?



a) 23

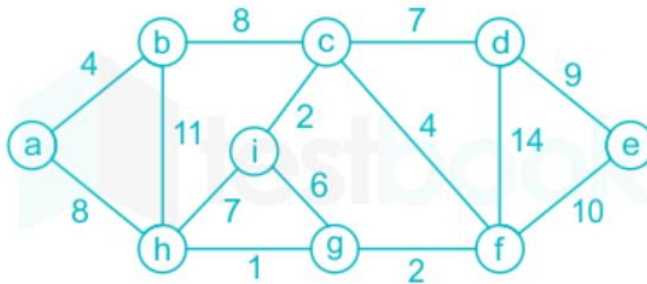
National University of Computer and Emerging Sciences  
Islamabad Campus

- b) 28
  - c) 27**
  - d) 11
2. In its standard implementation, which of the following sorting algorithms is not stable?
- a) Insertion sort
  - b) Merge sort
  - c) Counting sort
  - d) Quicksort**
  - e) Bubble sort
3. If every element in the input array is the same, which sorting algorithm will take the least amount of time? Examine usual methods that sorting algorithms are implemented.
- a) Insertion sort**
  - b) Merge sort
  - c) Counting sort
  - d) Quicksort
  - e) Bubble sort
4. The sorting method with the lowest worst-case complexity among the following is which one?
- a) Merge sort**
  - b) Insertion sort
  - c) Quicksort
  - d) Bubble sort
5. Insertion sort, Merge sort, and Quick sort have the following worst-case running times, respectively:
- a)  $\Theta(n \log n)$ ,  $\Theta(n \log n)$  and  $\Theta(n^2)$
  - b)  $\Theta(n^2)$ ,  $\Theta(n^2)$  and  $\Theta(n \log n)$
  - c)  $\Theta(n^2)$ ,  $\Theta(n \log n)$  and  $\Theta(n \log n)$
  - d)  $\Theta(n^2)$ ,  $\Theta(n \log n)$  and  $\Theta(n^2)$**
6. A sorting technique is called stable if:
- a)  $O(n \log n)$  time is required.
  - b) It preserves the non-distinct items' relative order of occurrence.**
  - c) It employs the divide and conquer strategy.
  - d) Auxiliary data storage space is not required.
7. In  $O(n)$  time, which algorithm sorts  $n$  numbers with the range 0 to  $(n^2-1)$  in ascending order with less space complexity?
- a) Insertion sort

National University of Computer and Emerging Sciences  
Islamabad Campus

- b) Merge sort  
**c) Radix sort**  
d) Quicksort  
e) Counting sort
8. It is assumed that the algorithms under consideration arrange the input sequences in ascending order. Which of the following statements is TRUE if the input is already in ascending order?
1. Quicksort runs in  $\Theta(n^2)$  time
  2. Mergesort runs in  $\Theta(n)$  time
  3. Insertion sort runs in  $\Theta(n)$  time
- b) I and II are correct  
c) II and III are correct  
**d) I and III are correct**  
e) I, II and III are correct

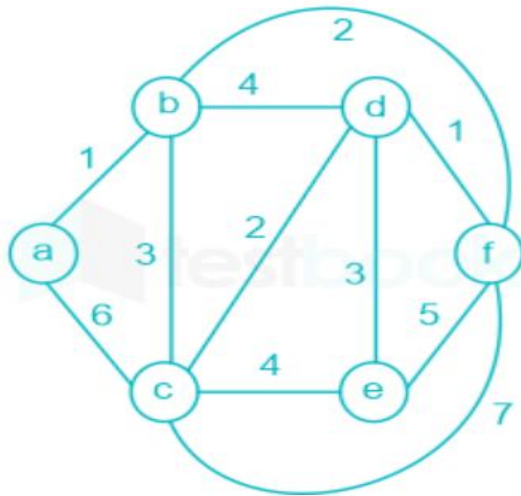
9. Consider following graph. Which of the following possible edges form the MST of the given graph using Prim's algorithm, starting from vertex a.



- a) (a, b), (b, h), (g, h), (g, i), (c, i), (c, f), (c, d), (d, e)  
**b) (a, b), (b, c), (c, i), (c, f), (f, g), (g, h), (c, d), (d, e)**  
c) (a, b), (g, h), (g, f), (c, f), (c, i), (f, e), (b, c), (d, e)  
d) None of the above
10. Based on the best-case time complexity of the following algorithms, which is the correct order?
- a) Merge sort > Quick sort > Insertion sort > selection sort  
**b) Insertion sort < Quick sort < Merge sort < selection sort**  
c) Merge sort > selection sort > quick sort > insertion sort  
d) Merge sort < Quick sort < selection sort < insertion sort
11. Which sorting technique works best for arrays with more than a million entries generally?
- a) Merge sort.  
b) Bubble sort.  
**c) Quick sort.**  
d) Insertion sort.

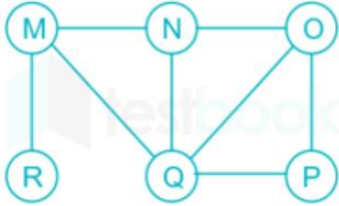
National University of Computer and Emerging Sciences  
Islamabad Campus

12. Sorting a list L, which is made up of a sorted list followed by few "random" elements, is your task. Which sorting technique would work best for this kind of task?
- a) Bubble sort
  - b) Merge sort
  - c) Quick sort
  - d) **Insertion sort**
13. Regarding comparison-based sorting algorithms, which of the following is not true?
- a) The minimum possible time complexity of a comparison-based sorting algorithm is  $O(n(\log(n)))$  for a random input array
  - b) Any comparison-based sorting algorithm can be made stable by using position as a criterion when two elements are compared
  - c) Counting Sort is not a comparison-based sorting algorithm
  - d) **Heap Sort is not a comparison-based sorting algorithm.**
14. Given the graph below which one of the following edges cannot be added in that order to find a minimum spanning tree algorithm using Kruskal.



- A (a-b)
  - B (d-f)
  - C (b-f)
  - D (d-c)
  - E (d-e)
- a) A, B, C, D, E
  - b) A, B, D, C, E
  - c) **B, A, C, E, D**
  - d) B, A, D, C, E
15. Regarding insertion sort, which of the following statements is true, where "Online" indicates that the algorithm has the ability to sort a list at runtime?

National University of Computer and Emerging Sciences  
Islamabad Campus

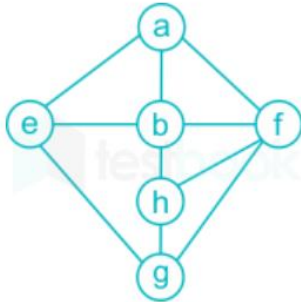
- a) **Insertion sort is stable, online but not suited well for a large number of elements.**
- b) Insertion sort is unstable and online
- c) Insertion sort is online and can be applied to more than 100 elements
- d) Insertion sort is stable & online and can be applied to more than 100 elements
16. What does it mean that the auxiliary space of insertion sort is  $O(1)$ ?
- a) The amount of memory (space) needed to process the data varies.
- b) It takes only 1 kb of memory.
- c) It is the speed at which the elements are traversed.
- d) **It means the amount of extra memory Insertion Sort consumes doesn't depend on the input. The algorithm should use the same amount of memory for all inputs.**
- e) All of the above
17. The Breadth First Search (BFS) algorithm has been implemented using the queue data structure. Which one of the following is a possible order of visiting the nodes in the graph below?
- 
- a) MNOPQR
- b) NQMPOR
- c) QMNROP
- d) **POQNMR**
18. When sorting a string of ASCII letters, which sorting algorithm is the most effective?
- a) Insertion sort
- b) Merge sort
- c) **Counting sort**
- d) Quicksort
- e) Bubble sort
19. Which sorting technique may be used to sort an array of integers in linear time if the numbers are between 1 and  $n^6$ ?
- a) Counting Sort
- b) **Radix Sort**
- c) Bucket Sort
- d) Quick Sort

National University of Computer and Emerging Sciences  
Islamabad Campus

20. Let  $G$  be a graph with  $n$  vertices and  $m$  edges. What is the tightest upper bound on the running time of Depth First Search on  $G$ , when  $G$  is represented as an adjacency matrix?

- a)  $\Theta(n)$
- b)  $\Theta(n + m)$
- c)  **$\Theta(n^2)$**
- d)  $\Theta(m^2)$

21. Consider the following graph



Which are the correct sequences for DFS

- 1. a b e g h f
  - 2. a b f e h g
  - 3. a b f h g e
  - 4. a f g h b e
- a) I, II, and IV only
  - b) I and IV only
  - c) II, III, and IV only
  - d) **I, III, and IV only**

22. Consider the graph  $M$  with 3 vertices. Its adjacency matrix is shown below. Which of the following is true?

$$M = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}$$

- a) Graph  $M$  has no minimum spanning tree
- b) Graph  $M$  has a unique minimum spanning trees of cost 2
- c) **Graph  $M$  has 3 distinct minimum spanning trees, each of cost 2**
- d) Graph  $M$  has 3 spanning trees of different costs

23. What is QuickSort's worst-case recurrence relation, and what is the time complexity in Worst case?

- a) Time complexity is  $O(n^2)$  and recurrence is  $T(n) = T(n-2) + O(n)$ .
- b) **The time complexity is  $O(n^2)$  and the recurrence is  $T(n) = T(n-1) + O(n)$ .**
- c) Time complexity is  $O(n \log n)$ , and recurrence is  $T(n) = 2T(n/2) + O(n)$ .

National University of Computer and Emerging Sciences  
Islamabad Campus

d) Time complexity is  $O(n \log n)$ , and recurrence is  $T(n) = T(n/10) + T(9n/10) + O(n)$ .

24. Assume that after completing the first partitioning, we are using quicksort to sort an array of eight integers, and the array now looks like this:

12 15 11 17 19 120 110 111

Which of the following statements is true?

a) **Either the 17 or the 19 might be the pivot.**

b) The pivot is 17, not 19

c) The pivot is 19, not 17

d) The pivot is neither the 17 nor the 19

25. Assume we have an algorithm that determines the median of an unsorted array in linear time  $O(n)$ . Consider a modified QuickSort implementation in which the median is used as a pivot once it has been determined using the aforementioned approach. What is this modified Quicksort's worst-case time complexity?

a)  **$O(n \log n)$**

b)  $O(n^2 \log n)$

c)  $O(n^2)$

d)  $O(n (\log n)(\log n))$

26. The  $(n/4)$ th smallest element is chosen as a pivot using an  $O(n)$  time technique in quick sort, which sorts  $n$  elements. What is the quick sort's worst-case time complexity?

a)  **$O(n \log n)$**

b)  $O(n^2 \log n)$

c)  $O(n^2)$

d)  $O(n (\log n)(\log n))$

27. Using the first element as a pivot, let  $Q$  be a QuickSort program that sorts numbers in ascending order. For the inputs  $\{1, 2, 3, 4, 5\}$  and  $\{4, 1, 5, 3, 2\}$ , let  $t_1$  and  $t_2$  be the number of comparisons that  $Q$  made, respectively. Which of the following is true?

a)  $t_1 = 5$

b)  $t_1 < t_2$

c)  **$t_1 > t_2$**

d)  $t_1 = t_2$

28. In the analysis of Prim's algorithm, what is the time complexity for the **Decrease key** operation using an **Fibonacci heap** as the priority queue?

a)  $O(v)$

b)  $O(\log V)$

c)  **$O(1)$**

National University of Computer and Emerging Sciences  
Islamabad Campus

- d)  $O(V.V)$
29. There are  $n$  elements in your array. Assume that the pivot is always the array's middle member while implementing quick sort. Then, for the worst-case performance, the tightest upper bound is
- a)  **$O(n^2)$**
  - b)  $O(n\log(n))$
  - c)  $\Theta(n\log(n))$
  - d)  $O(n^3)$
30. Which data structure is primarily used in Kruskal's algorithm to manage the connected components?
- a) Stack
  - b) Queue
  - c) **Disjoint-set**
  - d) Binary Search Tree
31. What is the time complexity of Prim's algorithm when using an **array** as the priority queue?
- a)  $O(v)$
  - b)  $O(\log V)$
  - c)  $O(1)$
  - d)  **$O(V.V)$**
32. You are working on a graph with 10,000 vertices and only 15,000 edges (a sparse graph). You need to find a minimum spanning tree using Prim's algorithm. Which data structure would be most efficient for the priority queue?
- a) Array
  - b) **Binary heap**
  - c) Linked List
  - d) Fibonacci Heap
33. Suppose you are implementing Prim's algorithm for a real-time system that requires frequent updates to edge weights (decrease-key operations are common). Which data structure would be ideal to handle these updates efficiently?
- a) Array
  - b) Binary Heap
  - c) Linked List
  - d) **Fibonacci Heap**
34. Take a look at an array-based binary max-heap implementation. Out of the arrays below, which one is a binary max-heap?
- a) **25,12,16,8,10,13,14**



National University of Computer and Emerging Sciences  
Islamabad Campus

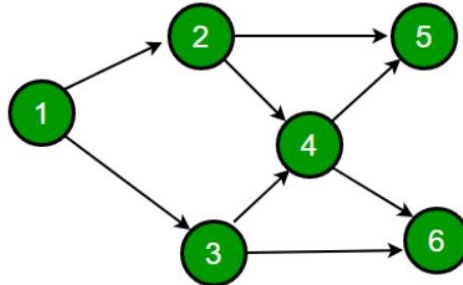
- b) 25,12,16,13,10,8,14
  - c) 25,14,13,16,10,8,12
  - d) 25,14,12,13,10,8,16
35. A min-Heap is a complete binary tree.
- a) **True**
  - b) False
36. Any DFS of the same graph will have at least one back edge if the DFS of the directed graph has one.
- a) **True**
  - b) False
37. What necessary condition must a tree have in order to be a heap?
- a) Only the tree must be complete.
  - b) Every Root value should be greater or smaller than the children's value only
  - c) **The tree must be complete and Every Root value should be greater or smaller than the children's value.**
  - d) None of the above
38. Out of the following, which is a valid heap?
- a) 16 14 10 4 7 9 3 2 8 1
  - b) **16 14 10 8 7 9 3 2 4 1**
  - c) 16 14 10 8 1 9 3 2 4 7
  - d) 16 14 3 4 1 9 10 8 2 7
39. There are \_\_\_\_\_ nodes of height h in any n-element heap.
- a) h
  - b)  $2^h$
  - c) Ceiling ( $n/2^h$ )
  - d) **Ceiling ( $n/2^{h+1}$ )**
40. A binary-max heap is given. The elements are kept in arrays with the following numbers: 25, 14, 16, 13, 10, 8, 12. After two delete operations, what is the array's content?
- a) 14,13,8,12,10
  - b) 14,12,13,10,8
  - c) **14,13,12,8,10**
  - d) 14,13,12,10,8
  - e) 16, 13, 10, 8, 12
41. Bucket sort is most efficient in the case when \_\_\_\_\_
- a) the input is non-uniformly distributed
  - b) **the input is uniformly distributed**

National University of Computer and Emerging Sciences  
Islamabad Campus

- c) the input is randomly distributed
  - d) the input range is large
42. What is the worst case running time of Rabin Karp Algorithm?
- a)  $\Theta(n)$
  - b)  $\Theta(n-m)$
  - c)  **$\Theta((n-m+1)m)$**
  - d)  $\Theta(n \log m)$
43. What is the pre-processing time of Rabin and Karp Algorithm?
- a)  $\Theta(m^2)$
  - b)  $\Theta(m \log n)$
  - c)  **$\Theta(m)$**
  - d)  $\text{Big-Oh}(n)$
44. Which algorithm among the following may be applied most effectively to identify whether a cycle is present in a given graph?
- a) **Depth First Search**
  - b) Breadth First Search
  - c) Prim's Minimum Spanning Tree Algorithm
  - d) Kruskal' Minimum Spanning Tree Algorithm
45. What are the appropriate data structures for following algorithms?
- 1. Breadth-First Search
  - 2. Depth First Search
  - 3. Prim's Minimum Spanning Tree
  - 4. Kruskal' Minimum Spanning Tree
- a) Stack, Queue, Priority Queue, Union Find
  - b) **Queue, Stack, Priority Queue, Union Find**
  - c) Priority Queue, Union Find, Stack, Queue
  - d) Priority Queue, Union Find, Queue, Stack
46. How can you determine if a graph contains cycle using DFS?
- a) **By checking if there is a back edge during traversal.**
  - b) By counting the number of vertices.
  - c) By counting the number of edges.
  - d) By checking if graph is not disconnected.
47. What is the meaning of it when In a graph if  $e=(u, v)$ ?
- a) u is the predecessor and v is the successor
  - b) e begins at u and ends at v
  - c) u is adjacent to v but v is not adjacent to u

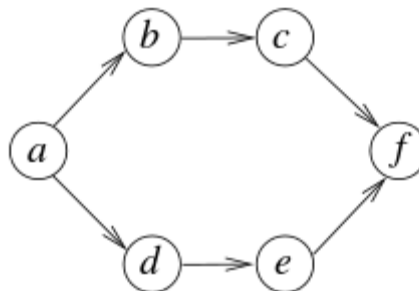
d) **Both a and b**

48. Examine the DAG below, where Consider  $V = \{1, 2, 3, 4, 5, 6\}$ . What is NOT a topological ordering among the following?



- a) 1 2 3 4 5 6
- b) 1 3 2 4 5 6
- c) 1 3 2 4 6 5
- d) **3 2 4 1 6 5**

49. Consider the following directed graph. The number of different topological orderings of the vertices of the graph is?



- a) 1
- b) 2
- c) 4
- d) **6**
- e) None of the above

50. Prefix function or pi value of pattern ACACAGAAACA is

- a) 01010012301
- b) 00122011122
- c) 00123012321
- d) **00123011123**
- e) None of the above

# National University of Computer and Emerging Sciences

## Islamabad Campus

**Q2:** Attempt following string matching problem.

[20 marks]

**Part a:** Create the PI tables for the following patterns according to the Knuth-Morris-Pratt (KMP) algorithm. Marks will only be awarded if the tables are fully correct.

[2 x 3 = 6]

P	a	b	a	a	b	a
$\pi$	0	0	1	1	2	3

P	a	b	a	b	a	c	a
$\pi$	0	0	1	2	3	0	1

P	a	b	c	a	b	a	b
$\pi$	0	0	0	1	2	1	2

**Part b:** Create transition table, finite automata machine for the given pattern P and traverse the text T on it and write the states passed. Write each state clearly and mentioned the shifts on which pattern is matched.

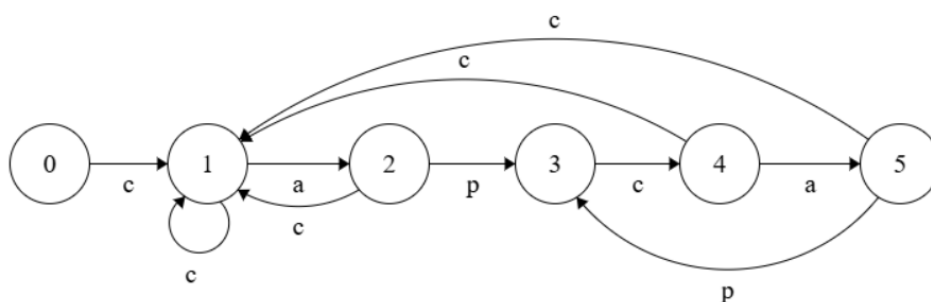
P = capca, T = acacapacpcaaapaacapcaap

[14 marks]

### Transition Table

	a	c	P
0	0	1	0
1	2	1	0
2	0	1	3
3	0	4	0
4	5	1	0
5	0	1	3

### Automata Machine / FSM



National University of Computer and Emerging Sciences  
Islamabad Campus

**Text Traversal**

<b>Index</b>	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
<b>Text</b>	a	c	a	c	a	p	a	c	a	p	c	a	a	a	p	a	a	c	a	p	c	a	a	p
<b>State</b>	0	1	2	1	2	3	0	1	2	3	4	5	0	0	0	0	0	1	2	3	4	5	0	0

**Valid Shifts: 7 and 17**

# National University of Computer and Emerging Sciences

## Islamabad Campus

### Important:

- Mark your answers to the MCQ's in the following answer sheet by FILLING the correct box. Tick or cross will not be marked. Try to fill the whole box.
- Any answers not provided here will not be marked.
- Multiple filled-in choices will be marked incorrect.
- In the Roll No column, use 6 digits of your registration number (ignoring alphabet); for example, 19i-8954 will be filled as in the right column.

<div style="text-align: center;">Roll No</div> <div style="border: 1px solid black; width: 100px; height: 20px; margin: 0 auto;"></div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">0</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">1</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">2</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">3</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">4</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">5</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">6</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">7</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">8</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">9</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="text-align: center; margin-top: 10px;">Subject 1</div> <div style="text-align: center;">Section 1</div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">1</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">2</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">3</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">4</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">5</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">6</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">7</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">8</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">9</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">10</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">11</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">12</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">13</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">14</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div>	<div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">15</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">16</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">17</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">18</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">19</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">20</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">21</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">22</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">23</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">24</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">25</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">26</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">27</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">28</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">29</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">30</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">31</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">32</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">33</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">34</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">35</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">36</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">37</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">38</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">39</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">40</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div>	<div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">41</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">42</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">43</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">44</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">45</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">46</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">47</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">48</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">49</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 15%;">50</div> <div style="width: 85%; text-align: center;">○ ○ ○ ○ ○ ○ ○</div> </div>
--	---	---