**Backend Task**

Data structure

Write a pseudo code implementing Queue DS using Arrays.

1. Give time complexity for Enqueue and Dequeue operations.
2. Given that each element has a priority and we need to implement a Max Priority queue, outline the changes you’ll make to the above implementation and give expected time complexity.

Database

Given table

* Employee -> {id, salary}; id is the primary key

Write an SQL query to report the second highest salary from the Employee table. If there is no second highest salary, the query should report NULL.

**Example** :

|  |  |
| --- | --- |
| id | salary |
| 2 | 150 |
| 5 | 300 |
| 6 | 200 |

**Output**: 200

|  |  |
| --- | --- |
| id | salary |
| 3 | 150 |

**Output**: NULL

SELECT TOP 1 CASE

WHEN MAX(salary) < (SELECT MAX(salary) FROM Employee)

THEN salary

ELSE NULL

END as Maximum

FROM Employee GROUP BY salary

Algorithms

**Give a pseudo code, time complexity, space complexity for each**

Given the head of a singly linked list, return *the middle node of the linked list*. If there are two middle nodes, return the second middle node.  
The number of nodes in the list is in the range [1, 100].

**Examples**:

**Input:** head = [1,2,3,4,5]

**Output:** [3,4,5]

**Explanation:** The middle node of the list is node 3.

**Input:** head = [1,2,3,4,5,6]

**Output:** [4,5,6]

**Explanation:** Since the list has two middle nodes with values 3 and 4, we return the second one.

Answer :

First after create the nodes (head) of the linked list we can divide the length of the linked list which will be divided over 2 if the result has a value > the length /2 we use the next node as a middle value location

Concerning with the total length of the list the equation will be

Result =Total length / location counter

If result > Total length / location counter then the middle will be the next node of the result

System

Develop an application, where a user can Post a story to their timeline. Story has a title and body. Each user can see other users’ timeline where they can find their stories' history. Another feature is to allow users to review a post, giving it a rate out of 5 and a comment (mandatory). Additionally users can see top posts, rated by average rate.

**Requirements:**

1- API to add a post

2- API to List User Posts with pagination

3- API to List Top Posts with pagination

4- API to add a review to Post, make sure that multiple users can add a review to the same post at the same time.

5- Test cases for the system with coverage for the parts you see are critical.

6- Seed database with 50k posts, more than 20k reviews.

7- No need for authentication or much user details, just a table with  id and username.

8- Provide ERD.

Make sure to have all requests below 100ms on an average machine.