

DS5003 Data Engineering Lab

Final Exam, Date: Nov 15, 2025

Timing: 10:00 to 13:00 PM

Max Marks: 30

Instructions

1. Submit one .ipynb file containing all answers for DSA questions and one .pdf file containing all the answers for the DBMS questions. The name should be [student name].ipynb and [student name].pdf respectively.
2. For the DBMS section you have to copy the query statement and paste a screenshot of your output below it.
3. The submission files cannot be > 2 MB in size, you should reduce screenshot resolution where needed.
4. Write the questions in separate text blocks before the answers.
5. Write justifications for your choices where needed.

1. Given an unsorted array of integers **nums**, find the length of the longest consecutive elements sequence. You must write an algorithm that runs in $\mathbf{O(n)}$ time. (3)

Example:

Input: nums = [100, 4, 200, 1, 3, 2]

Output: 4

2. Given a string **s** and an integer **k**, find the length of the longest substring that contains at most **k distinct characters**. (4)

3. You are given an array of meeting time intervals, where each interval is represented as [start, end]. You need to find the minimum number of meeting rooms required so that no two meetings overlap. (4)

Example:

Input: arr = [[0, 30], [5, 10], [15, 20]]

Output: 2

4. Write a Python function that exponentiates a square matrix of any dimension by diagonalization. You have to find eigenvalues and eigenvectors using Numpy for your solution. The steps to be used are as follows: (4)

1. Take a square matrix **A** as input
2. Find the eigenvalues (a vector **l**) and eigenvectors (a matrix **V**) of **A**
3. Apply the Python function `exp()` to the components of **l** and form a diagonal matrix **L** with those values on the diagonal

4. Multiply the resulting matrix L with the eigenvector matrix from both sides, in this way: $V L V^{-1}$
5. The result is the exponential of the matrix A; that is, $e^A = V L V^{-1}$

Demonstrate your function's working for the following matrices:

$$M = [[1., 0., 0.], [0., 2., 0.], [0., 0., 3.]]$$
$$N = [[1., 2., 3.], [3., 2., 1.], [4., 5., 6.]]$$

Create a dvdrental database in postgres. Run the following from the command line to set up the dvdrental database:

```
pg_restore -U postgres -d dvdrental dvdrental.tar
```

5. Find the total amount paid by each customer along with their full name and email. Show only those customers who have made more than 5 payments. Sort the result by total amount in descending order. (4)
6. Create a view named **rental_details_view** that shows detailed information about each movie rental, including: (5)
 - rental_id
 - rental_date
 - film_title
 - customer_name
 - staff_name
 - store_id

Then display the top 5 most rented films along with the number of times each film was rented.

7. Write a SQL function to count the films that are in the language input by the user. Use this function to count the number of films in each language in the collection. (6)

DVD RENTAL ER DIAGRAM

