# CS5481: Data Engineering - Assignment2

## Instructions

1. Due at Tuesday, Oct. 31, 2023, 12:59:59 PM;
2. You can submit your answers by **a single PDF with the code and the output files** or **a jupyter notebook with output files** containing both the answers and the code;
3. For the coding questions, besides the code, you are encouraged to additionally give some descriptions of your code design and its workflow. Detailed analysis of the experimental results are also preferred;
4. Total marks are 100;
5. If you have any questions, please post your questions on the Canvas-Discussion forum or contact TA Mr. Wei Shao (email: weishao4-c@my.cityu.edu.hk).

# Question 1 - LLM for data engineering

(**18 marks**) LLMs' fast, articulate answers to expert questions can help data engineers discover datasets, write and debug code, document procedures, and learn new techniques as they build data pipelines. In this question, you are required to write suitable prompts for ChatGPT to achieve the following targets.

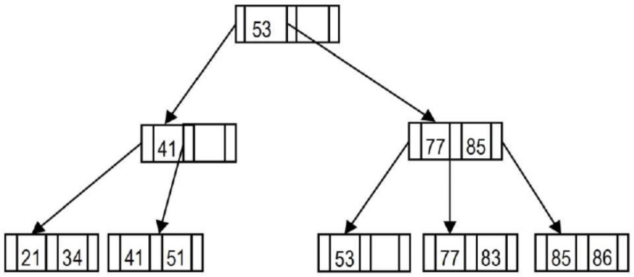
1. (4 marks) Assume you need some datasets for training a knowledge-grounded dialogue generation model. Please utilize the ChatGPT (GPT4) to find the datasets you need. List the prompts you give and the outputs of ChatGPT.

2. (6 marks) CMU_DoG dataset is a document grounded dataset for text conversations. "Document Grounded Conversations" are conversations that are about the contents of a specified document. In this dataset the specified documents are Wikipedia articles about popular movies. The dataset contains 4112 conversations with an average of 21.43 turns per conversation. Please use ChatGPT to preprocess a conversation sample. You should list the prompts you use and the inputs to ChatGPT and outputs from ChatGPT.

3. (8 marks) Let the ChatGPT generate a preprocessing code to process CMU_DoG dataset for the inputs of T5 model. List the prompts you use and the outputs of ChatGPT. Considering that errors in codes are inevitable, you should fix the codes to make it run. Please use the fixed codes to preprocess the conversation samples in CMU_DOG and list the first 5 processed conversation samples here.

# Question 2 - Data Indexing

(**25 marks**) Given the following $B^+$-tree, please answer following questions.




1. (5 marks) What is the value of p for this $B^+$-tree? (Note that p is the order of a $B^+$-tree)
2. (6 marks) Can you re-build a taller $B^+$-tree with the same value of p using the same set of search-key values in the leaf nodes of the given tree? If yes, show the steps by drawing a new diagram whenever the height of the tree increases.
3. (6 marks) Insert the search-key values 32, 84, and 19 in sequence to the given $B^+$-tree, and draw a new diagram for each insertion.
4. (8 marks) Suggest a sequence of search-key values to be deleted from the resultant $B^+$-tree in Q4.2 to shrink the tree to 2 levels with the **least** number of deletions. Show the steps by drawing a new diagram whenever a node is deleted.

# Question 3 - Data Querying

(**27 marks**) The university conducted a hacker contest which consists of a set of small tasks. The participants can submit the solutions to each task and obtain the score regarding each submission. The top-3 participants of each task will win the corresponding bonus.

Given the following SQL tables,

```
Hackers(hacker_id: INT, name: VARCHAR, Bank_account: INT)
Tasks(task_id: INT, task_description: VARCHAR, bonus: INT)
Submissions(submission_id: INT, hacker_id: INT, task_id: INT, score: INT, submission_data: DATE)
```

Assuming that,

- The id values, including `hack_id`, `task_id`, `submission_id`, are unique and serve as foreign keys.
- There are 10,000 hackers registering the contest, but not all hackers have the submissions.
- There are totally 50 small tasks. For each task, hackers can have multiple submissions, but only the last submission obtaining the best score will be considered when calculating the final winner shortlist. If more than three submissions achieved top-3 scores, only the earliest three submissions will be rewarded. The earlier submission has the smaller `submission_id`.
- There are totally 100,000 submissions. The scores of all tasks range from 0 to 100.

1. (**5 marks**) Write the query to print the `hacker_id`, `name` of the hacker who has the most submissions. If more than one such hacker has the maximum submission number, the results are sorted by ascending `hack_id`.
2. (**5 marks**) Write the query to print the `task_id`, `task_description` and `the number of submissions received` of the task which receives the most submissions. If there are more than one such task, the results are sorted by the ascending `task_id`.
3. (**5 marks**) Write the query to print the `submission_id`, `name`, `task_description` of the submission whose score is the highest one among all scores obtained on 2022-10-01. If more than one such submissions, the results are sorted by ascending `submission_id`.
4. (**5 marks**) Write the query to print the `hacker_id`, `name` of the hacker who has the highest total scores on all tasks. For each task, if the hacker has multiple submissions, only the best score is accounted. If more than one such hacker, the results are sorted by descending `hacker_id`.
5. (**5 marks**) Write the query to print the `hacker_id`, `name`, `bonus` of the hackers and the task fulfilling the requirement that the hackers win the bonus on task with `task_id=25`.
6. (**2 marks**) Write the query to print the `hacker_id`, `hacker_name`, `bank_account` and `the total bonus` of all participants. The result is sorted by descending `total bonus` and ascending `hacker_id`. Exclude all hackers with a total bonus of 0 from the result.

# Question 4 - Recommender System

(**30 marks**)

1. **(8 marks)** Please write two basic approaches for recommender system and briefly explain them.
2. **(8 marks)** Cold start is a severe problem for recommender system. Please explain what is cold start problem in recommender system and how to solve it (list at least TWO methods).
3. **(14 marks)** Rating prediction is an important task for the recommender system. Try to implement a recommendation model on the Movielens-100k dataset to predict user rating. There are some instructions as following:
   (a) You can download the dataset from https://files.grouplens.org/datasets/movielens/ml-100k.zip
   (b) You just need to train on `ua.base` and evaluate on `ua.test`.
   (c) You can use methods such as matrix factorization, two tower models, or graph-based models.
   (d) For basic requirements, you can only use the rating file in the dataset. To obtain richer features of user/item, you can also use the corresponding user/item information in the dataset.

**Reference**

[1] Huang, Po-Sen, et al. "Learning deep structured semantic models for web search using click through data." Proceedings of the 22nd ACM international conference on Information & Knowledge Management. 2013.

[2] Koren, Yehuda, Robert Bell, and Chris Volinsky. "Matrix factorization techniques for recommender systems." Computer 42.8 (2009): 30-37.

[3] Wang, Xiang, et al. "Neural graph collaborative filtering." Proceedings of the 42nd international ACM SIGIR conference on Research and development in Information Retrieval. 2019.