CS 412: Introduction to Data Mining	Fall 2020

<p style="text-align:center">Homework 1</p>

Handed Out: Feb 19, 2020 12:00 am *Due: Mar 06, 2020 11:59 pm*

1 General Instructions

- The programming assignment will be hosted on hackerrank (`https://www.hackerrank.com/`) as a programming contest. To participate in this contest, please open a hackerrank account with your illinois.edu email netid. If you already have a username in hackerrank that is different from your netid, please fill out your netid and username in this google form (`https://forms.gle/bWSeytDD54nEdWBN8`).

- It is OK to discuss the problems with the TAs and your classmates. However, it is NOT OK to work together or share code. You need to write your code independently and the TAs will not do the code debugging. Plagiarism is an academic violation to copy, to include text from other sources, including online sources, without proper citation. To get a better idea of what constitutes plagiarism, consult the CS Honor code (`http://cs.illinois.edu/academics/honor-code`) on academic integrity violations, including examples, and recommended penalties. There is a zero tolerance policy on academic integrity violations. Any student found to be violating this code will be subject to disciplinary action.

- Please use Piazza first if you have questions about the homework. Also feel free to send us e-mails and come to office hours.

2 Programming Assignment Instructions

This question aims to provide you a better understanding of the frequent pattern mining and the closed/maximal pattern mining. Participate in the programming contest hosted at hackerrank: `https://www.hackerrank.com/cs412-hw1`.

- Please read the problem description carefully.

- The input will always be valid. We are mainly testing your understanding of frequent pattern mining, not your coding skills.

- Please pay special attention to the output format. We will be using the hackerrank based autograder and it is extremely important that your generated output satisfies the requirement.

- We don't have specific constrains for this programming question. The only constrains are the standard environment constraints in hackerrank: `https://www.hackerrank.com/environment`.

- The grading will be based on how many test cases you passed. You are provided with two sample test cases to debug your code. For the final grading, we will use additional test cases to test your code.

- Please use Piazza first if you have questions about the homework.

3   Programming Assignment (50 points)

This question aims to provide you a better understanding of the frequent pattern mining and the closed/maximal pattern mining.

1. Implement a frequent pattern mining algorithm (e.g., the Apriori algorithm or FP-Growth) to mine the frequent patterns from a transaction dataset.

2. Implement a closed pattern mining algorithm to mine the closed frequent patterns from the same transaction dataset. An easy way is to write code based on the frequent patterns you got from part 1.

3. Implement a maximal pattern mining algorithm to mine the maximal frequent patterns from the same transaction dataset. An easy way is to write code based on the frequent patterns you got from part 1.

Input Format

The input dataset is a transaction dataset.

The first line of the input corresponds to the minimum support.

Each following line of the input corresponds to one transaction. Items in each transaction are seperated by a space.

Please refer to the sample input below. In sample input 0, the minimum support is 2, and the dataset contains 3 transactions and 5 item types (A, B, C, D and E).

Constraints

NA

Output Format

The output are the frequent patterns you mined out from the input dataset.

Each line of the output should be in the format:

```
Support [frequent pattern]
Support [frequent pattern]
......
```

The frequent patterns should be ordered according to their support from largest to smallest. Ties should be resolved by ordering the frequent patterns according to the alphabetical order.

First print all the frequent patterns for part 1, then the closed frequent patterns for part 2 and last the maximal frequent patterns for part 3. Each part should be separated by an empty line.

Please refer to the sample output below. In sample output 0, the first 9 patterns are the frequent patterns for part 1, the following 3 patterns are the closed frequent patterns for part 2 and the last 2 patterns are the maximal frequent patterns for part 3.

Sample Input 0

```
2
B A C E D
A C
C B D
```

Sample Output 0

```
3 [C]
2 [A]
2 [A C]
2 [B]
2 [B C]
2 [B C D]
2 [B D]
2 [C D]
2 [D]

3 [C]
2 [A C]
2 [B C D]

2 [A C]
2 [B C D]
```

Sample Input 1

```
2
data mining
```

frequent pattern mining
mining frequent patterns from the transaction dataset
closed and maximal pattern mining

Sample Output 1

4 [mining]
2 [frequent]
2 [frequent mining]
2 [mining pattern]
2 [pattern]

4 [mining]
2 [frequent mining]
2 [mining pattern]

2 [frequent mining]
2 [mining pattern]