

Course name: Data Science (ITE4005)

Professor: Sang-Wook Kim (email: wook@agape.hanyang.ac.kr)

TAs: DongHyuk Seo (email: hyuk125@agape.hanyang.ac.kr)

Jiwon Son (email: tinybeing@agape.hanyang.ac.kr)

< Programming Assignment #1 >

3 Mar. 2021

Due Date: 23 March 2021, 11:59 pm

1. Environment

- OS: Windows, Mac OS, or Linux
- Languages: C++, Java, or Python (any version is ok)

2. Goal: find association rules using the **Apriori** algorithm

3. Requirements

The program must meet the following requirements:

- Execution file name: apriori.exe
- Execute the program with three arguments: minimum support, input file name, output file name
 - Example:

```
C:\#>apriori.exe 5 input.txt output.txt
```

- Minimum support = 5%, input file name = 'input.txt', output file name = 'output.txt'
- If you python, you are allowed to use 'apriori.py' file instead of 'apriori.exe'

- Input file format (.txt)

`[item_id]\t[item_id]\n`

`[item_id]\t[item_id]\t[item_id]\t[item_id]\t[item_id]\n`

`[item_id]\t[item_id]\t[item_id]\t[item_id]\n`

- Row: transaction
- `item_id` is a numerical value
- There is no duplication of items in each transaction

- Example:

18	2	4	5	1	
1	11	15	2	7	16
2	1	16			
15	7	6	11	18	9
11	2	13	4		

- Output file format Figure 1. Input file example (.txt)

`[item_set]\t[associative_item_set]\t[support(%)]\t[confidence(%)]\n`

`[item_set]\t[associative_item_set]\t[support(%)]\t[confidence(%)]\n`

- `[item_set]\t[associative_item_set]`: association rules with minimum support
 - `[item_set]→[associative_item_set]`
 - Use braces to represent item sets: `{[item_id],[item_id],...}` (*Important!!*)
 - e.g., `{0}`, `{0,4}`, `{0,3,1}`
- *Support*: probability that a transaction contains `[item_set] ∪ [associative_item_set]`
- *Confidence*: conditional probability that a transaction having `[item_set]` also contains `[associative_item_set]`
- The order of output is unimportant.
- The value of support and confidence should be rounded to two decimal places.
 - e.g., 24.631 rounded to two decimal places should become 24.63.
- An additional penalty will be imposed if you don't keep the output file format.
- Example:

<code>{1}</code>	<code>{8}</code>	15.40	51.68
<code>{8}</code>	<code>{1}</code>	15.40	34.07
<code>{1}</code>	<code>{9}</code>	9.60	32.21
<code>{9}</code>	<code>{1}</code>	9.60	34.53
<code>{1}</code>	<code>{10}</code>	10.20	34.23
<code>{10}</code>	<code>{1}</code>	10.20	35.17

- **Note: Please make sure to match the output format!**
If the format is not correct, you can't get any score.

4. Submission

- Please submit the program files and the report to GitLab
- Report Figure 2. Output file example
 - The file format of report must be *.pdf.
 - Guideline
 - ✓ Summary of your algorithm
 - ✓ Detailed description of your codes (for each function)
 - ✓ Instructions for compiling your source codes at TA's computer (e.g. screenshot) (*Important!!*)
 - ✓ Any other specification of your implementation and testing
- Program files
 - A executable file (.exe or .py)

- All source files
 - ✓ MakeFile if you use Linux
- Note: submission details for GitLab will be announced later.

5. Penalty

- Late submission
 - 1 week delay: 20%
 - 2 weeks delay: 50%
 - Delay more than 2 weeks: 100%
- Requirements unsatisfied
 - Significant penalty up to 30% will be given when the requirements are not satisfied