Course name: Data Science (ITE4005)

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

TAs: Jangwan Koo (email: koojwan@agape.hanyang.ac.kr)
Tae-ri Kim (email: taerik@agape.hanyang.ac.kr)

< Programming Assignment #1 >

5 Mar. 2018

Due Date: 28 March 2018, 11:59 pm

1. Environment

- OS: Windows, Mac OS, or Linux
- Languages: C, C++, 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'
- 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
```

Figure 1. Input file example

- Output file format (.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:

{0}	{1}	6.60	24.63
{0,1}	{2}	7	25
{7}	{0,4}	3.4	14
{0}	{3}	2.2	10
{0,1}	{2}	4	17
{0,1}	{3}	1	4

Figure 2. Output file example

4. Submission

- Please submit the program files and the report to GitLab
  - Report
    - Should be written in *English*
    - The file format of report must be *.docx, *.doc, *.hwp, *.pdf, or *.odt.
    - 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)
    - 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