Audiobook recommender system

1. Introduction

Let us assume, that there is an online audiobook store that displays common properties for each audiobook offered for sale. Such properties include the ratings given by previous buyers, a summary, name of the author, narrator, and length. Users can rate audiobooks on a five-point scale. Based on these reviews, the system recommends additional audiobooks to each user.

2. Assignment

Implement a recommendation algorithm in Java or Python! The algorithm can be chosen freely, but you are not allowed to use any code that you yourself have not written. The only exception in case of a Java implementation is the Apache Commons library (http://commons.apache.org), which can be used.

2.1. Java

The code must contain a *Main* class, and within this, a *main()* function. It will receive all inputs on the standard input, and should output the solution to the standard output. Upload the zipped source code files of your application to the BME MIT HomeWork portal. (https://hf.mit.bme.hu).

2.2. Python

The code must be a single python file, that will be run and receives all inputs onto the standard input, and it should write the solution to the standard output. Upload the zipped single python file to the BME MIT Homework portal. Use Python3.x, and only standard libraries are available (e.g. no numpy!) (https://hf.mit.bme.hu).

2.3. Input

The first line of input contains the number of known ratings, users and audiobooks. Then each line contains a user ID, an audiobook ID, and the corresponding rating. The fields are separated by a tab. Example:

```
60000   500   200
0       3     4
0       31    3
```

```
0       87      2

...

499     192     5

499     198     4
```

2.4. Output

The output contains the IDs of the top 10 recommended audiobooks for each user, separated by tabs.

Audiobook Identifiers should be ordered such that the top of the list, i.e. the most recommended audiobook, comes first. However, the list should not include an audiobook whose rating is known (already given by that user)! The rows should follow each other in the order of user IDs. These IDs should not be displayed.

In case of the example above, the output contains 500 lines:

```
175  21   76   77   119  2    40   42   56   117

32   142  38   111  14   18   101  138  64   29

75   88   47   43   18   150  83   124  166  182
```

3. Evaluation

The rating is based on the number of users for which there is an overlap between the list of top n recommended audiobooks and the list of actually top rated audiobooks.