Home work until October 25, 2016 (23h59):

Task 1: Data acquisition and basic analysis (6 pts)
Write a crawler for fetching Last.fm listening histories of users ("recent tracks"). To this end, elaborate a method to collect listening information for at least 500 users and at least 5,000 unique artist. You will likely need to combine various Last.fm API functions to create a user base.
Store the **listening histories (user-id/name, artist-id/name, track-id/name, timestamp)** in some structured form, e.g. text files, to ease further exploitation towards building a recommender system.
Apply basic *data cleansing* techniques (e.g., matching with metadata in a music database, removing artists and users who provide too little information/too sparse data).
In addition to the listening histories, also store **basic characteristics of the users** for whom you fetch listening histories (API function *user.getInfo*: country, age, gender, registered, total playcounts, etc.).

In your report, describe the method you designed for creating the user base (i.e., how did you select the 500+ users?) and the data cleansing techniques you implemented. Then, have a look at the paper describing the LFM-1b dataset:
http://www.cp.jku.at/people/schedl/Research/Publications/pdf/schedl_icmr_2016.pdf
and compare the characteristics of your dataset to those of the LFM-1b set, with respect to demographics (country, age, and gender). Can you make any interesting observations? E.g., is the country, age, or gender distribution different? If so, can this be explained by the method you designed for selecting users?

Task 2: Recommendation: collaborative filtering and random baseline (6 pts)
Extend the user-based, memory-based **collaborative filtering** artist recommender that we implemented in the lab, so that it considers more than just one nearest neighbor of the target user. When doing so, you have to combine the artists recommended by different neighbors, of course. Think about a clever way to do this, i.e., elaborate a **method to combine the predictions for the same artists among the set of nearest neighbors** (e.g., how to deal with an artist that is recommended 10 times by 20 nearest neighbors vs. an artist that is recommended only once, but by a neighbor with a music taste very similar to the target user?). The two factors you may want to consider here are the number of neighbors who suggest a certain artist and the similarity of the neighbors to the target user.

Similarly to the `recommend_RB` function created in the lab, implement a random baseline recommender that randomly picks a user and recommends his artists, not listened to by the target user.

In the report, describe your extension to the CF recommender, in particular how you create the eventual recommendations by combining the recommendations by individual nearest neighbors. Also provide a code snippet of your random baseline recommender.

Task 3: Evaluation (3 pts)
Using the provided dataset C1ka (on Mediacube), which is a subset of the LFM-1b set, perform **10-fold cross-fold validation** on the user level. Evaluate the CF and the random baseline recommendation algorithms you implemented and compute average **precision, recall, and F1 measure**.
Report these performance measures for the two recommenders and different numbers of artists (in case of RB, e.g., 1, 5, 10, 20, 50, 100) and number of neighboring users (in case of CF, e.g., 1, 2, 3, 5, 10, 20).

Do not forget to name all team members in your report!