

- All solutions may be submitted in groups of up to 3 students.
- Send your solutions before the due date per email to `cloudcomputing@uni.lu` in a single zip file (containing just the java files or scripts) using your group id as the file name.
- Include some brief instructions on how to run your solution to each problem in a file called `problem_X.txt`.
- All group members are required to be present in the class for the demonstration of the solutions on the due date.

III. HBase & MongoDB

JOIN OPERATION IN HBASE

8 Points

Problem 1. Modify the `NestedLoopJoin.java` example given on the moodle to find:

- all the subjects (x) and objects (y and z) matching the pattern: `?x <hasGivenName> ?y. ?x <livesIn> ?z.`, from the Yago dataset.
- the family name of all the persons who are citizens of more than two countries. You may use the predicates: `<isCitizenOf>` and `<hasFamilyName>`, for mining the pattern.

Use the `yago2.tsv` available on the course homepage on `moodle.uni.lu` for this problem. Bulk-load data from the Yago dataset into the required tables to perform the join operation.

AGGREGATION PIPELINE AND MAP-REDUCE IN MONGODB

14 Points

Problem 2. Perform the two queries mentioned in Problem-1 using:

- Aggregation Pipeline
- Map-Reduce

Consider again the `yago2.tsv` for this problem. For case (i), you may use `$lookup` along with other pipeline operators to construct the query. Please refer to `Ex3.P2.hints.txt` file on the moodle for hints. For case (ii), you may follow the Reduce-Side join example illustrated in the lecture to perform the join operation.

- Compare the runtime obtained in both these cases (i.e, Problem-2 (i) and (ii)).
- Consider creating indexes on various fields of the collection and observe how the runtime changes.