Students have to solve the programming tasks (using Java) in groups of 4-5 students. Please give a name to your group and email your group name, student names, Immat. Number by 10th Nov '19 EOD to vikram.apilla@ovgu.de. At least 50% of all programming points are needed in order to qualify for the written exam. There will be a total of 2 programming tasks with an overall total of 20 points. Prog. Task 01 will have 15 points, Prog. Task 02 will have 5 points. If you achieve 10 or more points in task 01, then you do NOT need to submit the second task.

# Programming Task P01

Program your own (command-line based) Information Retrieval system using *Apache Lucene*[1] (at least version 3.6, currently the newest version is 8.2.0, Recommended Java version is 7). *Lucene* is an open source search library that provides standard functionality for analyzing, indexing, and searching text-based documents. The following criteria have to be met by your Information Retrieval system. Your program shoud ...

- Using Lucene, parse and index *Plain Text* and *HTML* documents that a given folder and its subfolders contain. List all parsed files.
- Consider the English language and use a stemmer for it (e.g. Porter Stemmer).
- Select an available search index or create a new one.
- Print a ranked list of relevant articles given a search query. The output should contain the most relevant documents, their rank, path, last modification time, relevance score and in addition for *HTML* documents title and summary.
- Search multiple fields concurrently: not only search the document's text (body tag), but also its title and date (for *HTML* documents).

The program should be written in a way that it is runnable without taking any look in the source code or even adapting the source code. Create a jar-File named IR_P01.jar. It should process the input:

```
java -jar IR_P01.jar [path_to_document_folder]
```

Please send your solutions (jar-File AND source code and a short documentation of 3-6 pages( explaining the code structure and important aspects) via e-mail until 29. Nov. 2019, 23:59 to vikram.apilla@ovgu.de.

The subject of the email should be: IRP01 [Group-Name] Submission

[1] http://lucene.apache.org/java/

- 5 points: for the short implementation documentation (min 3 pages, max 6 pages) explaining the flow of the overall program, how it is implemented, what is the overall logic, which sections of the code cater to functional sections like parsing, stemming, indexing, searching
- 5 points: how is the actual source code is structured? Tip: 0 points for an implementation where the entire code resides in a single file. You need to divide the logic in smaller blocks and structure them in a logical manner. This code structuring should be explained in the documentation.
- 2 points: if the code runs without any change and achieves at least one of the above mentioned features (like parsing, stemming, indexing, ranked-search).
- 3 points: if the all of the mentioned features work 100 percent correctly!
- Plagiarism of any kind will lead to 0 points of 15 points. If you 'borrow' the code, you need to cite it clearly in the implementation documentation. You need to explain in simple terms what is precisely done by that piece of code!

If you are really unlucky and cannot find colleagues to form groups for the programming task, email to vikram.apilla@ovgu.de by 8th Nov 2019 EOD. We will create groups out of the applications received by random allocation. Mention the subject of the email as: [IRP01-Cannot find group]. Exclusions: There can be no change of group members once the allotment is done. There will be no chance to swap group members for Prog. task 2, after the group for Prog. task 1 is finalized.

***(15 points)***