***Report:-***

***Indexer:***

***Indexer.py*** *contains python code to create inverted indexes. It reads the document set from* ***tccorpus.txt****, computes inverted indexed, converts it to JSON object and stores that object in a file named* ***index.out.json****.*

***Implementation:-***

***Indexer.py*** *read the* ***tccorpus.txt*** *line by line and stores the tokens corresponding to each document in a list of dictionaries called* ***doc\_tokens\_list****. Further the program iterates over token and documents in* ***doc\_token\_list*** *to compute inverted index. Inverted indices are stored as dictionary of dictionaries having a structure* ***{‘token’: {doc\_id: frequency}}****. Further, inverted index is wrapped as JSON object and written to a JSON file. This JSON file is later consumed by* ***bm25.py*** *to implement BM25 algorithm.*

***Bm25:***

***Bm25.py*** *contains python code to read index.out.json file and implement BM25 algorithm.*

***Implementation:-***

***Bm25.py*** *reads* ***index.out.json*** *and converts it to equivalent inverted index dictionary. This inverted index dictionary is later used in computation of following values and data structures:*

***N****: No. of documents in tccorpus.txt.*

***Doc\_len****: A dictionary having structure* ***{doc\_id: length}.*** *It contains length of each document.*

***Avdl****: Average document length.*

***Doc\_dl\_avdl\_ratio****: A dictionary* ***{doc\_id: dl/avdl}*** *containing* ***dl/avdl*** *ratio of each document.*

***Document\_term\_freq****: A dictionary* ***{query\_term: n}*** *that contains document term frequency for each term in the given queries.*

*Further* ***bm25.py*** *reads* ***queries.txt*** *and computes query term frequency for each term in the given set of queries. All these values and data structures are further used to implement BM25 algorithm. BM25 score for each document in* ***tccorpus.txt*** *with respect to all the given queries ae computed and stored in* ***result.eaval.txt*** *in the format specified in the problem set.*