**CSC 555: Final**
**Winter 2024**
**Due Date/End Date**
**03/22/24-03/22/24 5PM CT**

---

This exam contains 3 pages (including this cover page) and 2 problems. Check to see if any pages are missing.

**Submission Format:**

- **Please make sure to submit all python code.** If you are using Jupyter notebooks, please export it into the main document or into a .py file before submission. **No Jupyter notebook must be submitted. Only .py files.**
- Every student must submit the following statement along with their exam in a simple text file: I have done this exam entirely on my own. I have not consulted with friends or consulted online resources that have the solution to the exam.

**Plagiarism and Collaboration Policy:**

- This is a single person exam. Collaborations are not allowed.
- I have created a *moderated* Exam Discussion Board. Students can post, but it will be moderated by me before being visible. If I respond, I may respond to you by email or allow the post to be visible by everyone. If I do not respond, assume that you are asking for part of a solution that I cannot divulge.
- Use of Web is only permitted to seek Python documentation help. We know all sources where correct and incorrect solutions to this exam lie. So do not attempt to search for those.

**Cluster, Discussion Board, Web use:**

- For this exam the use of Hadoop cluster (2 or more nodes) is optional. In other words, no extra points will be awarded or deducted for using/not using the cluster. However, everyone should be able to run on a single node Hadoop installation. A medium to large instance will suffice.
- I have created a moderated Exam Discussion Board. Students can post, but it will be ratified by me before being visible. Depending upon the question, I may or may not respond. If I respond, I may respond to you by email or allow the post to be visible by everyone. If I do not respond, assume that you are asking for part of a solution that I cannot divulge.
- Use of Web is only permitted to seek Python documentation help. We know all sources where correct and incorrect solutions to this exam lie. So do not attempt to search for those.

1. **Clustering**

   Write Python code within the four files provided. Each file has instructions to write code as described in the Python triple quotes, i.e., "'TODO:..."'. You must write code using these files and not write your own files.

   To write code you will need to be familiar with numpy. The tutorial at this site `https://numpy.org/doc/stable/user/quickstart.html` will be sufficient.

   Once you have generated data, generated the centers, and written the code in all four files, test it with:

   ```
   cat kmeans_data.csv | python3 mapper-kmeans.py | sort -n | python3 reducer-kmeans.py > centers.txt
   ```

   Now repeat the process and run 4 iterations of kMeans. Remember to change the centers file in mapper-kmeans.py everytime to correspond to the centers generated in the previous iteration.

   Note, you can maintain centers as `centers[i].txt` where [i] is the centers output by the $i^{th}$ iteration. The first centers file must be generated apriori and is used by mapper-kmeans.py

   Submit the following:

   - The initial centers.txt and the kmeans_data.csv. Please zip the kmeans_data.csv.
   - All four Python files.
   - The final centers.txt after 4 iterations.
   - The Hadoop command used and a screenshot of running on Hadoop cluster with time information for each iteration.

2. **BloomFilter-based 2-way map-side Join** The objective of this question is to implement a 1-pass MR that uses bloom filter for the larger table.

   Write Python code within the three files provided. Each file has instructions to write code as described in the Python triple quotes, i.e., "'TODO:..."'. You must write code using these files and not write your own files.

   Consider the following query:

   ```
   select *
   from lineorder, dwdate
   where lo orderdate = d_datekey
   and d_sellingseason = 'Fall'
   ```

   Lineorder and Dwdate are to be downloaded from `http://cdmgcsarprd01.dpu.depaul.edu/CSC555/SSBM1/dwdate.tbl`
   `http://cdmgcsarprd01.dpu.depaul.edu/CSC555/SSBM1/lineorder.tbl`

   In the map phase, the bloom filter of the large table (lineorder) will be read. The small table (dwdate) will also be read. In the map phase, make both the where clause checks and output the join result.

   The reduce phase just passes the result.

   The bloomfilter should be setup as described in bloomfilter.py.

   You can test code as:

```
cat Dwdate.tbl | python3 BFmapper.py | sort -n | python3 BFreducer.py
```

where the mapper.py reads the stored bloomfilter.

Submit the following:

- All three Python files.
- The Hadoop command used and a screenshot of running on Hadoop cluster with time information.
- The result of your join as a text file.