**Pseudocode**

Hadoop Mapper and Reducer for Pairs-Approach

1.a mapper\_pairs.py

---------------------------

for each line in sys.stdin

Split the line on ‘\t’ to get userID, movieID and rating

If rating>=4

Store the userID and corresponding movieID as key-value pairs inside dictionary

For each userID in the dictionary

Create a movie list, sorted by the movieID’s

for i in range ( 0, len ( sorted\_movielist ) ) :

for j in range ( i + 1, len ( sorted\_movielist ) ) :

if i != j:

emit movieID1, movieID2, 1

1.b reducer\_pairs.py

----------------------------

Read the output from mapper

For each line of mapper

Split the line on ‘\t’

Sum the count values to the corresponding pairs in dictionary

For each pair in dictionary

Split the pair on ‘,’

If pair in dictionary has the count > 200

emit movieID1, movieID2, count

Note: for the data sets Size 1m and 10m, the ratings.dat file has separated field values as “::”. So in mapper, we split the values on the “::” to get the userID, movieID, rating

**For matching the movieID’s with the movie names**

Movie\_names.py

-----------------------------

Open the file u.item by importing the file in python program

Give the movieID’s along with the count as an input

For each line in movieFile u.item

Split the line on ‘|’ to get movieID and movie\_name

Store the movieID and movie\_name as key-value pairs

If rating > 200

Emit movie\_names for the corresponding movieID’s along with their counts as values