**Pseudocode**

Stripes Approach in Spark

Spark\_stripes\_100K.py

Read the input file u.data by taking it in SparkContext

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

If rating>=4

Make pairs of userID and movieID and send it to map function

Reduce the output by calling reduce function

Take the movie tuple(movie stripes) and then reduce it by using reduceByKey function which will give the corresponding count

Emit co-occurring movies along with their frequency

Function map1:

Takes the userID and movieID pairs and assign it to a list(array)

Return this array

Function reduce1:

For each user ID

Make the movie pairs groupBy userID

Generate the movie stripes along with the count

Generate the co-occurrence dictionary(associative array) and sum up the counts

For every key in dictionary, split the pairs and store the movie names along with their counts in the dictionary

Return this dictionary

Function loadMovieNames:

Load the file ‘u.item’ and split each line on ‘|’

Map the movieID’s with the corresponding names

Return movie name dictionary

For the data set 1m and 10m, we will split the ratings.dat file on ‘::’ and takes the ratings.dat file as the SparkContext