Project Report

Question 1

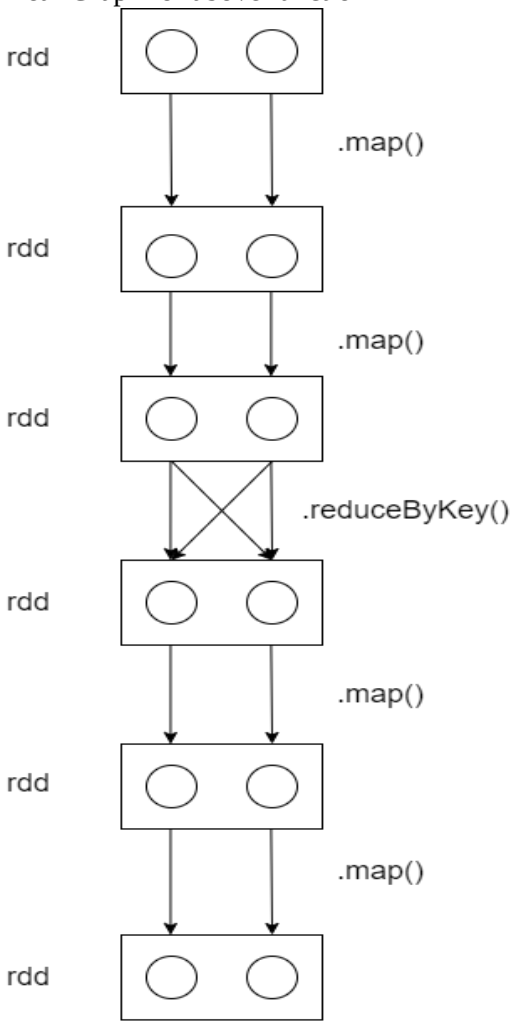
1.1. Function chosen Task 2 get monthly contacts' function.

```
def get_monthly_contacts(rdd):
from operator import add


rdd = rdd.map(lambda s: (str(s[0])+"-"+str(s[2].month)+'/'+str(s[2].year),s[1]))
rdd = rdd.map(lambda s: (s[0],1))
rdd = rdd.reduceByKey(add)
rdd = rdd.map(lambda s: (s[0].split("-"),s[1]))
rdd = rdd.map(lambda s: (s[0][0],s[0][1],s[1]))
return rdd
```

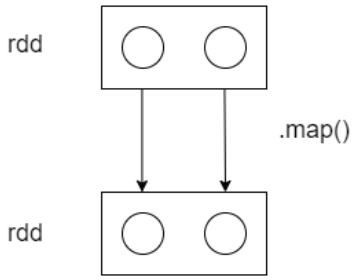
In this function, I have extracted month and year from datetime object and the problem we faced was to count the number of occurrences, so in order to deal with this - we joined the sender and the month and year in order to get the count by reduceByKey easily.

1.2. Linear Graph for above function



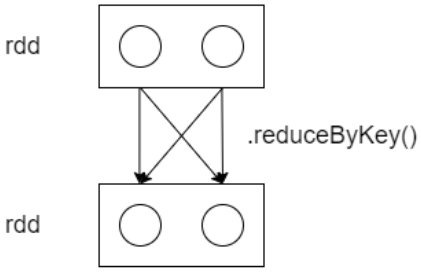
1.3. Narrow Dependencies.

Yes, there are narrow dependencies in the lineage graph.



1.4. Wide Dependencies.

Yes, there is wide dependency in the lineage graph.



1.5. Different stages and tasks of linear graph in 1.2

In this function there is just 1 stage. The tasks in the function are:

Task 1: Appending sender, month and year as one attribute and other as receiver.

Task 2: Initialising the first entry in the tuple with its initial respective counts as 1

Task 3: reduceByKey to count the number of key-value pair and merges the values with the same key.

Task 4: Splitting the first entry of the tuple into sender and mm/yyyy

Task 5: Removing the unnecessary list which was generated on the sender and recipient pair which was generated after split.