①

Ans1:

**Splitting**

**Shuffling**

input

| A → B C D |
| B → A C D E |
| C → A B D E |
| D → A B C E |
| E → B C D |

Splitting boxes:

A → BCD

B → ACDE

C → ABDE

D → ABCE

E → BCD

Shuffling boxes:

A,B → BCD, ACDE →

A,C → BCD, ABDE →

A,D → BCD, ABCE →

A,E → BCD, BCD →

B,C → ACDE, ABDE →

B,D → ACDE, ABCE →

B,E → ACDE, BCD →

C,D → ABDE, ABCE →

C,E → ABDE, BCD →

D,E → ABCE, BCD →

**Reduce**

Count the number of
occurrence of each friend
and keep the one having
count 2.

| A,B → C D |
| A C → B D |
| A D → B C |
| B C → A DE |
| B D → A CE |
| BE → CD |
| CD → ABE |
| CE → BD |
| DE → BC |

**final result**

| (AB) → (CD) |
| (AC) → (BD) |
| (AD) → (BC) |
| (BC) → (ADE) |
| (BD) → (ACE) |
| (BE) → (CD) |
| (CD) → (ABE) |
| (CE) → (BD) |
| (DE) → (BC) |

②

Assumption: input to reducing algorithm is a ˣ list of
combined

A, B, C, D and E's facebook friends list.

A,B list, A,C list, A D list, AE, B,C list, B D list, BE' list, CD list,

CE list, and DE's list

For each list

    class Mapper

        Method Map (list AB)

           for each element

                EMIT. (friend, count)

    class Reducer → methode reduce will ged list of
                      friends and each one cout on list.
       method Reduce

          method reduce (list friends-count)

             for all friends
             if (cout ==2)
               EMIT (friend, count)
             endif.
             else
                continue.

    end - for

→ after reducing phase you will get lists of
AB, AC, AD, AE, BC, BD, BE, CD, CF, DE lists having common friends

→ Sun up all the lists and show the resut in readable
format.

3. Spark Scala implementation:

pseudo code for implementation:

val textfile = spark.texfile ("~~txt~~ D:\ facebook_list.txt;

~~shuffile~~
val splits = textfile. ~~a~~fltmap ( ~~time~~ => ~~line~~.shuffile());

val common : ~~textfile. fled mp~~
   friends = splits. flatmap(line => line.split (" "))
              . map (word => (word, 1))
              . reduce By Key ( count >2)

commonfriends. save file (" D:\ common friends");

ReduceByKey : reduces the intermediate results by "count = 2"

shuffile function : creates AB, AC, AD, AE like wise combinations