For example the input is a <key, value> like (Isaac, [Mikel, John, Lucy]), which represent (one person, all friends of this person). After Map1, split the input list of values by keys separately, and swap the separated key-value pairs, which is a list of key-values: [(Mikel, Isaac), (John, Isaac), (Lucy, Isaac)], and this represent (one of the person's friend, the person mentioned in the input). After Reduce1, we concatenate all the same keys together, and for example, one of the output can be (Mikel, [Isaac, John, Lucy, Claudia]), which is also one of the input in Map2. During Map2, We break apart each value in the list of values from input and pair them up in pairs to form a new key for an output value, while the original keys in the input become the output values, so in this example, we will obtain ([Isaac, John], Mikel), ([Isaac, Lucy], Mikel), ([Isaac, Claudia], Mikel), ([John, Lucy], Mikel), ([John, Claudia], Mikel), ([Lucy, Claudia], Mikel). This indicates the two person in keys have their common friends Mikel. In the Reduce2, we simply combine the values corresponding to each identical key. After this, we got the final result.