Data Science for Business Mid-term mini exam **Jannuary 2018**

Questions For each of the following questions indicate the letter corresponding to the right answer.

1. Mechanisms for robustness in Hadoop have been introduced
    A. in order to avoid that a Map/Reduce task fails
    B. in order to allow for re-execution of Map/Reduce tasks that have failed
    C. in order to allow for the fast re-execution of the entire MapReduce job in case of a failure
2. In the execution of a MapReduce job, whose input is a file partitioned and stored in 10 chunks by means of HDFS, the number of Map tasks
    A. must be equal to 10
    B. can be greater than or equal to 10
    C. can be either greater than 10 or less than 10 or equal to 10.
3. During the shuffle&sort process for a MapReduce job execution, couples that are returned by Map tasks
    A. are first grouped and sorted on computers running each Map task
    B. are grouped and sorted exclusively on machines running Reduce tasks
    C. are exclusively grouped and sorted on machines not running any Map/Reduce task of the job execution.
4. Consider an existing MapReduce program whose execution implies an excessive number of couples sent to the shuffle&sort phase. In order to decrease this number:
    A. the only possibility is to add a Combiner
    B. either a Combiner is added or the Map is rewritten
    C. the Reduce function could be rewritten

Exercise Given two multi-sets A and B, design in pseudo-code or Python a MapReduce job computing their set-difference: elements in A must be returned (only once, without repetition) in case they do not occur in B. Note that since A and B are multi-sets their elements may appear more than once in each of the multi-sets.
Assume that A is represented by a file A.txt where each element corresponds to a text line (with no space), and similarly for B. For instance we could have

A.txt including the following lines	B.txt including the following lines
aa bb aa cc aa bb hhh	aa ff cc dd dd

In this case the job results the lines bb and hhh. So bb, even if it occurs twice, in A.txt it is resulted only once. Do you believe that adding a Combiner could improve scalability? Why? (just motivate your answer with 3-4 text lines).

*D*esign in pseudo-code a MapReduce program performing multiset-difference: in this case an element occurring n times in A and m times in B, is resulted n-m times if n-m>0; the element is not resulted if n-m <=0 (observe that both n and m can be 0: occurring 0 times means not occurring). In the above example, the job would result 2=3-1 times the element aa, 2=2-0 times the line bb and 1=1-0 time the line hhh.