Exercise Sheet 2

Task 3

- The ~~master~~ document collection gets split into splits. The master has to ~~give the~~ information assign splits to the parsers. So the task description for the parsers have to contain which split(s) the parser shall ~~parse~~ work on.
- The parsers report back (term-docId) pairs as j-term partition with $j = \#\text{partitions}$.
- The task description of an inverter needs to include on which term-partition it has to work on.
- The inverters sort their postings list (~~containing of~~ consisting of (term, docId) pairs of one term-partition for all splits) and report them back.

Task 4

The idea of logarithmic merging is to have a index in memory that can hold up to n items. If it reaches the limit it will be written to the disk ~~However it is not always merged with all the (~~previously written~~~~ ~~indexes but only~~ or merged with a index that has already been written to the disk. If this is the case the next time it will be written as new index to the disk. Afterwards it will be merged with the next bigger written index. This way ~~the~~ we only merge with indexes of size $\log(n)$ and the merging takes less time making our system more responsive to the user than if we would have only one aux. index and one main index.

The merging of the aux index and the indexes on disk could be realised the same way the distributed indexing works. That way the merging process could be speeded up but the ~~search~~ querying during that time would be very hard to realise.