//JATIN LAMBA //A20362075

Answers:

Timing of jobs with MaxTemparatureWithCombiner for the files:

- 5085.txt: 2m58s
- 9285.txt: 5m44s

This process took a longer time when compared to others because here the splitting process does occur and due to which it takes some time to compress as we all know it's a fairly slow process.

After Intermediate Compression, the times are as follows:

5085.txt:

- Gzip: 2m15s
- Snappy: 1m51s
- LZ4: 1m59s

9285.txt:

- Gzip: 4m25s
- Snappy: 3m05s
- LZ4: 3m30s

Now, after the modification of code we can see that intermediate compression reduces the time as all of these codecs are fast enough to run jobs on big data as they don't support splitting which results in the faster process. Therefore, yes intermediate helps in reducing the performance timings.

80.txt:

- WithoutCompression: 74m44s
- Gzip: 39m29s
- Snappy: 30m52s
- LZ4: 33m30s

From the above output it is clear that Snappy codec is the fastest for compression but we can also see that when the job was run without the compression it took some time because it supports splitting and is fairly a slow process. Whereas, the rest codec doesn't support the splitting and end up in a faster job.