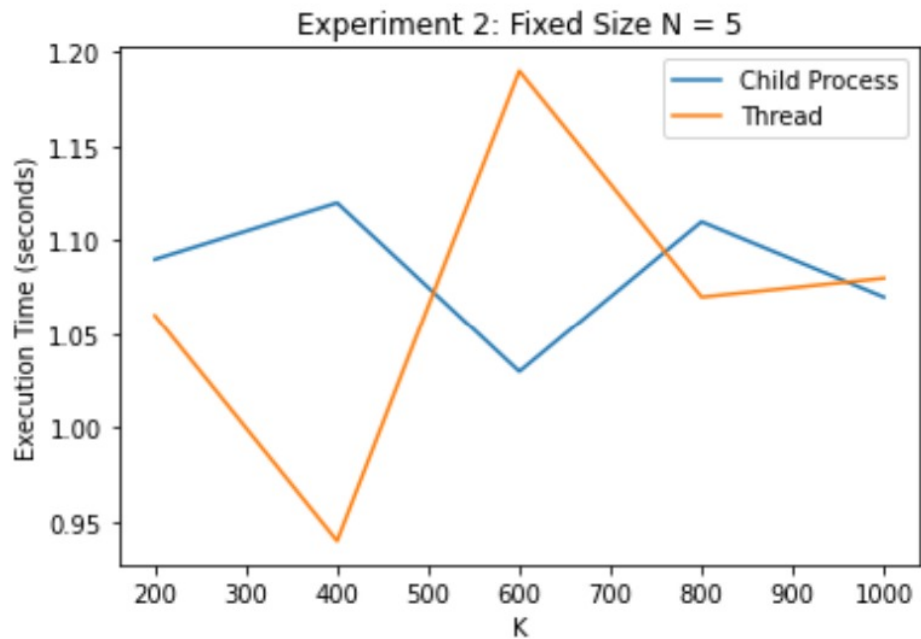


For Experiment 1:

- Both child process and thread implementations show a similar trend in terms of execution time.
- As the value of N increases, the execution time for both implementations increases as well. This is expected since there are more files to be processed as N increases.



For Experiment 2:

- Both child process and thread implementations show a similar trend in terms of execution time.
- The execution time for both implementations increases as the value of K increases. This is expected since the algorithm has to process a larger number of words to find the top K words.
- The child process implementation seems to have a slightly higher execution time compared to the thread implementation for higher values of K.

In general, the results are expected since both child processes and threads are used to parallelize the processing of input files. However, we can observe that the child process implementation seems to have a slightly lower execution time compared to the thread implementation. Overall, there is no significant conflict between the results of the two experiments.

It's also worth noting that the performance of the experiments is affected by the hardware used. In our case, we have an **Intel i7-9750H processor, 16 GB of RAM, and an NVMe2 SSD**. This CPU has **6 cores and 12 threads**, which can influence the performance of the program, the maximum number of size (N) is too low for this CPU since 6 processes will physically be able to run on the CPU simultaneously. This hardware configuration may differ in terms of the performance of the experiments compared to other configurations, so the results may not be generalizable to all hardware configurations.