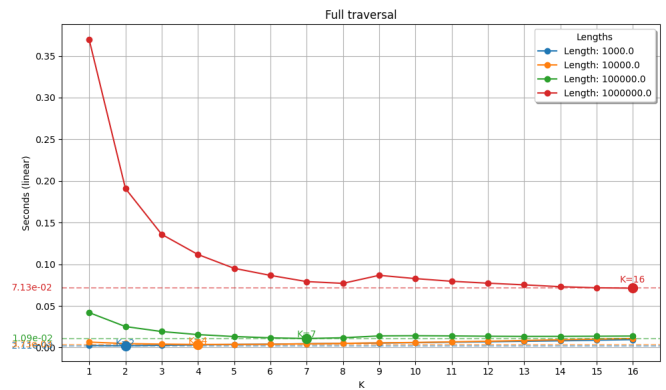
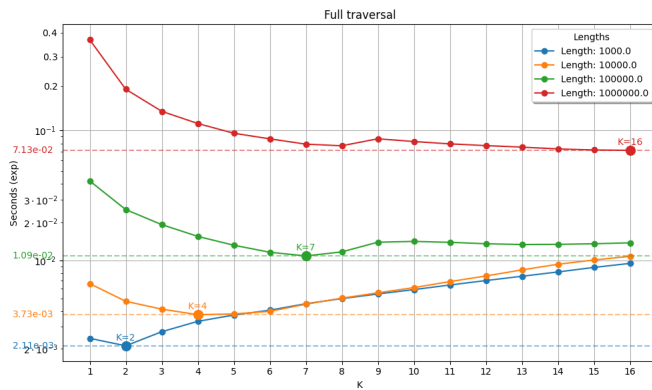
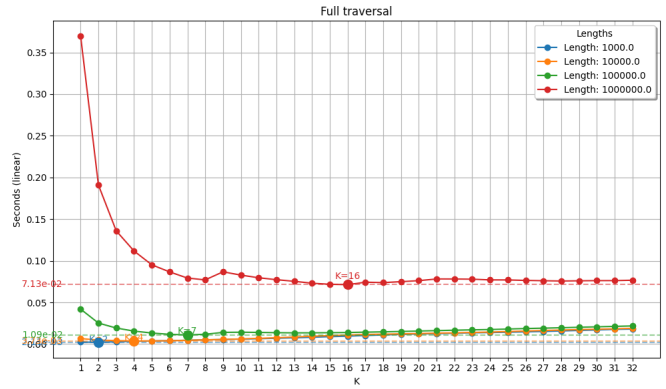
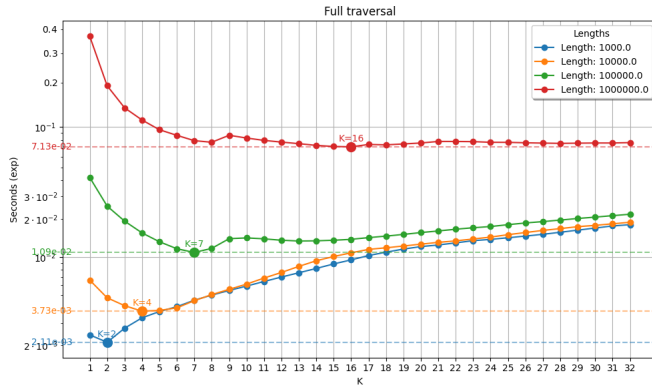
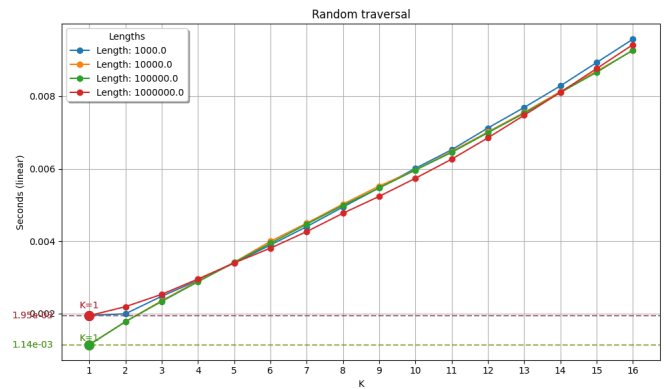
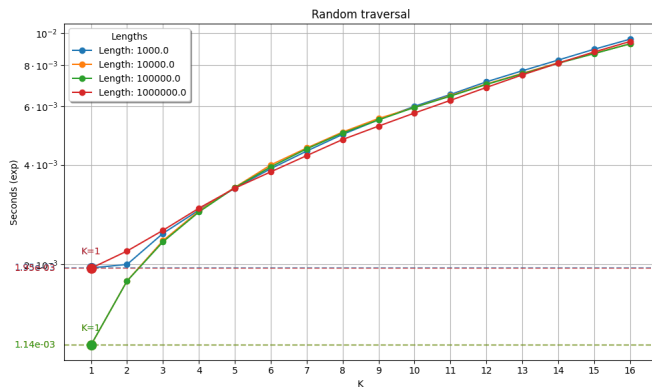
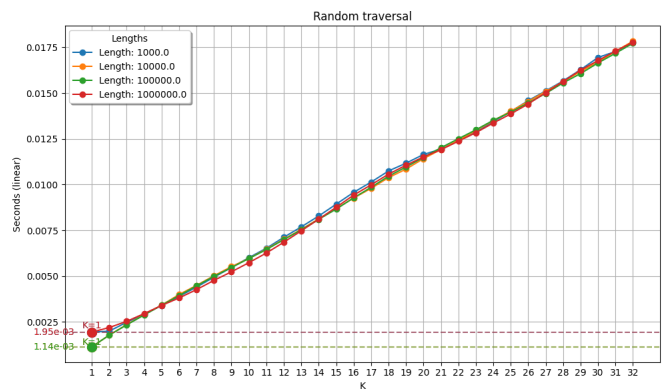
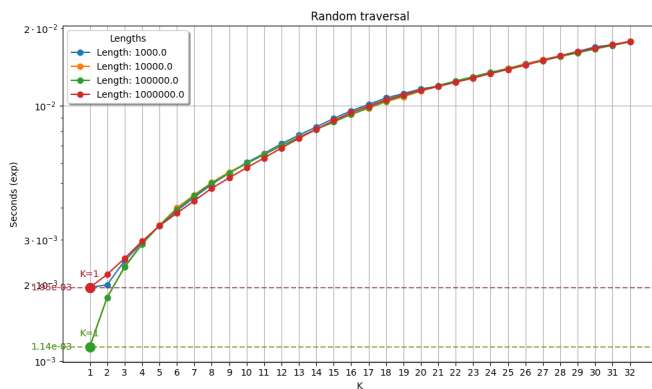


Звіт

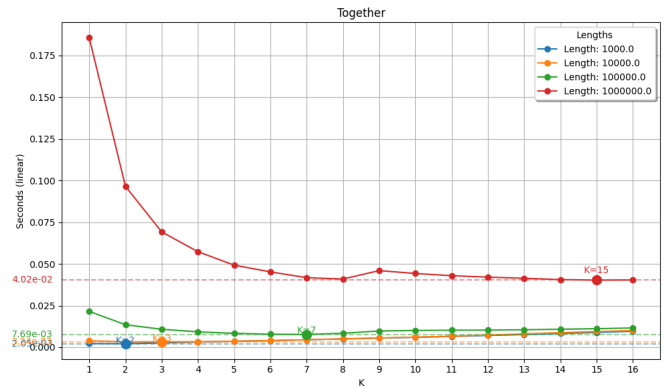
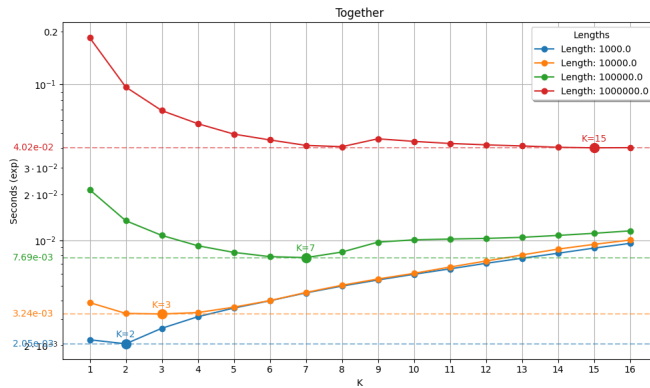
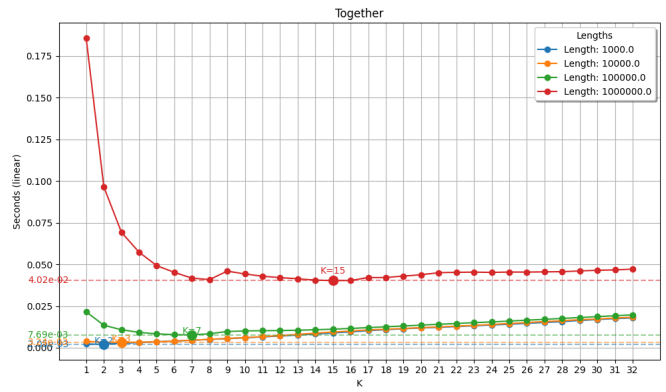
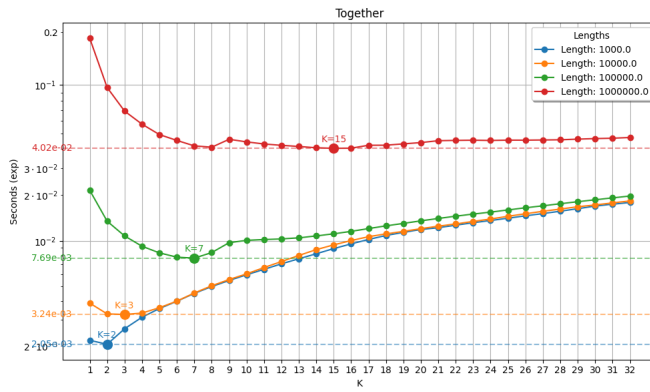
Графіки



Обхід послідовності, який відбувався завжди до самого кінця



Обхід послідовності, який міг завершитися у довільний момент



Загальний результат

Висновок

Дослідження показало, що найкраща швидкість у бібліотечного алгоритму **none_of** досягається при використанні 15-16 потоків (за необхідності повного обходу послідовності) та при використанні 1 потоку (за умови, що обхід може завершитися на будь-якому елементі).

Дослідження також показало, що використання більше ніж 16 потоків на процесорі, що має всього 16 потоків, є неефективним – є сенс триматися лише у межах від 1 до 16 потоків.

За умови повного обходу послідовності, за збільшення K час, необхідний на обхід послідовності, зростає за законом $\frac{1}{c_1 \cdot K^p} + c_2$, де c_1 та c_2 – деякі додатні константи, а $p > 0$ (тобто гіперболічно).

За умови, що обхід може завершитися на будь-якому елементі, за збільшення K час, необхідний на обхід послідовності, зростає за законом $c_1 \cdot K + c_2$, де c_1 та c_2 – деякі додатні константи (тобто лінійно).

Конкретні числові значення тривалості роботи алгоритму за різної кількості потоків (а також значення тривалості за відсутності та наявності різних політик) прикладаються до звіду у вигляді файлів **results.csv** та **results_random.csv**.

З вищезазначених файлів можна побачити, що за першої умови найкращою політикою є **par_unseq**, а за другої умови – **unseq**. Відсутність політики програє в обох випадках.