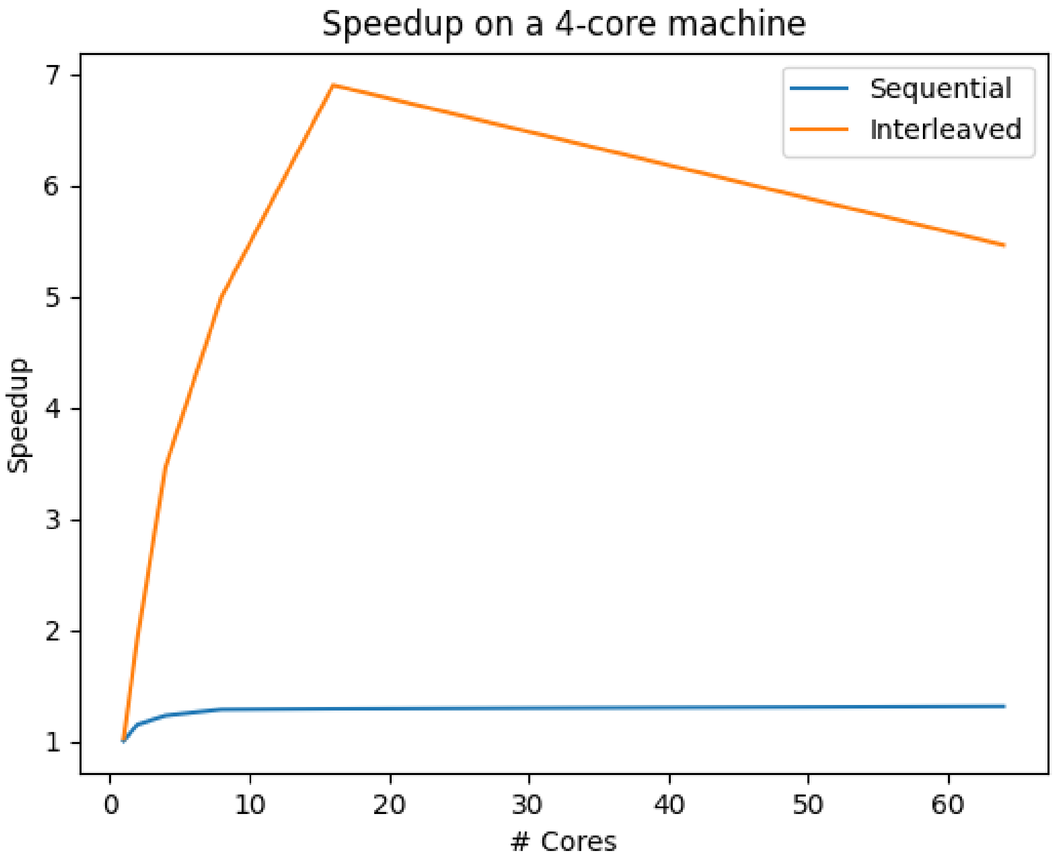
Ari ‘aith’ I.



In part 1, I converted an 'embarrassingly parallel' for-loop into two versions that allow either a computer's pipelining or superscalar architecture to process the for-loop in parallel. My first version involves loop unrolling s.t. multiple elements in the for-loop can perform their addition operations within the same for-loop iteration. However, this does not fix the data dependency between each addition of that element; each addition operation is a dependency for the immediately-proceeding addition operation.

Version 2 aims to fix this dependency chain by interleaving each element's addition operation. This allows even the dependent addition operations to be performed in parallel since they are not executed sequentially anymore.