Parallel and Concurrent Programming

Assignment 5

Irfan Asfia Ali - CS16BTECH11019

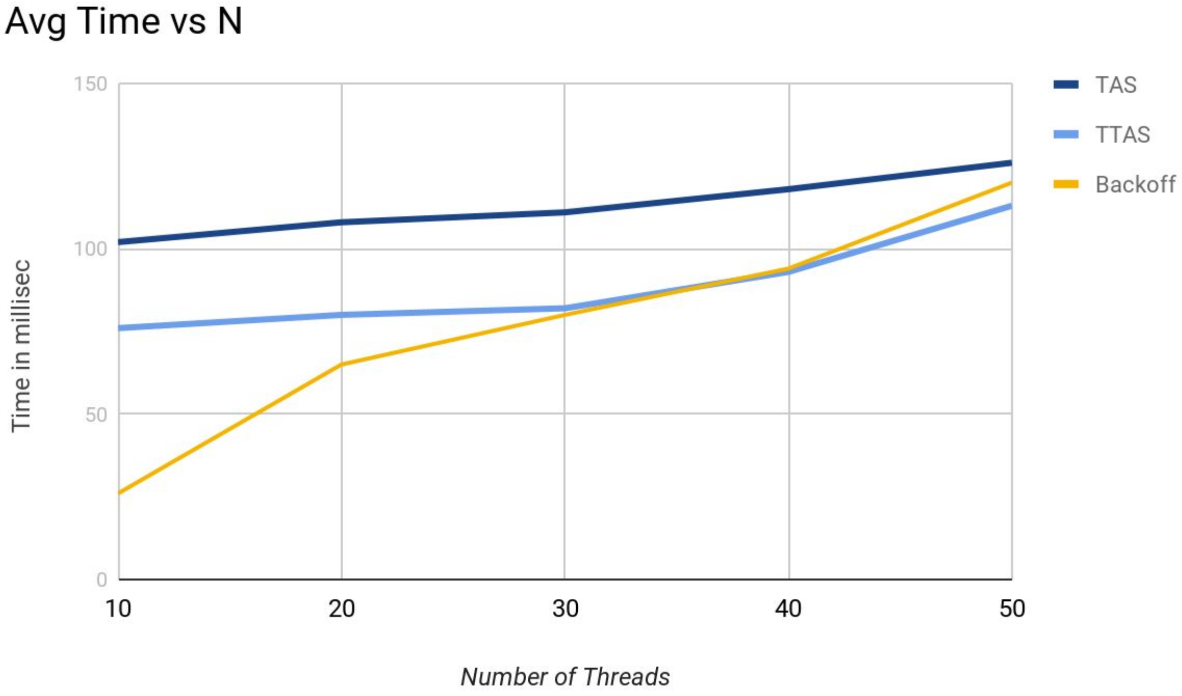
The Goal of the assignment was to compare the performance of three different types of locking mechanisms: TAS, TTAS and Backoff

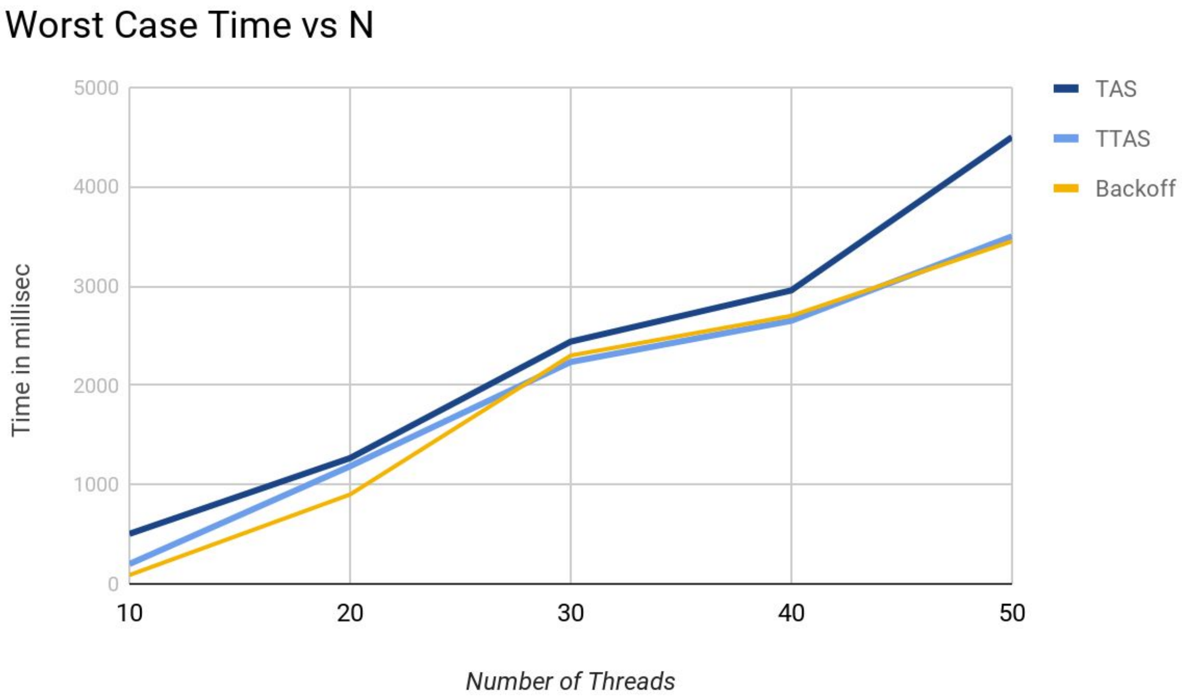
TAS Lock : TAS standing for testAndSet() uses an atomic boolean variable to check if the lock is free. Any thread accessing the lock uses getAndSet() method on the atomic variable until the method returns false. This method itself is atomic. This lock spins on the value of the boolean variable. The thread acquiring the lock sets the value of the boolean variable to true. While unlocking the thread sets the value to false.

TTAS Lock : Test-and-TestAndSet. In this locking method, instead of performing testAndSet() directly, the thread repeatedly reads the lock until it appears to be free. Once the thread finds that the lock is free, then it starts testAndSet. This lock performs much better than TAS but is still not near to ideal performance

Backoff Lock: In this method if a thread is trying to acquire a lock which seems to be having a high contention, then the thread takes a backoff before retrying and lets the competing threads get a chance to finish. The backoff time for a thread is chosen randomly in a range(min,max). In my case the min and max values of the lock are set to 1 and 32 microsecs respectively

The graphs below give the performance comparison.





We can see that TAS performs worse but TTAS and Backoff are better. TTAS and Backoff almost have similar performance, Backoff being a little better.