Benchmark Caveats

- All benchmarks are within a single process
- On an MacBook Pro M2 Max (with 8 cores, the number of P-cores)

```
template<typename queue_type>
void measure_serial(benchmark::State& state)
{
    const size_t queue_size = state.range(0);
    const size_t iters = 4 * queue_size;

    for (auto _: state)
    {
        // Initialise the queue half full
        queue_type queue(queue_size);

        for (size_t i = 0; i < (queue_size / 2); ++i)
            push (queue, i);

        stopwatch sw;

        for (size_t i = 0; i < iters; ++i)
        {
            if (! queue.try_push (i))
                continue;

            int v;
            if (! queue.try_pop (v))
                continue;

            benchmark::DoNotOptimize(v);
        }

        state.SetIterationTime(sw.get());
   }

    // Throughput is calculated as an item being enqueued and dequeued, so travelling fully through the queue
    state.SetItemsProcessed(int64_t(state.iterations()) * iters);
}
```