

**Q1.** How much faster does it run on a multiprocessor machine?

**Answer.** If we compare the result from the single-thread implementation to the optimal one, yields an  $\frac{134932524-19620621}{134932524} \approx 85.4\%$  increase in performance.

**Q2.** What is the optimal number of goroutines to use?

**Answer.** The optimal number is the number of `runtime.NumCPU()` or `nproc`, the performance plateaus after this number.

```
goos: linux
goarch: amd64
pkg: hw7
cpu: AMD Ryzen 5 7640U w/ Radeon 760M Graphics
BenchmarkSingleThread
BenchmarkSingleThread-12          8 134932524 ns/op
BenchmarkWorkerPool3
BenchmarkWorkerPool3-12          25 45646155 ns/op
BenchmarkWorkerPool5
BenchmarkWorkerPool5-12          39 28737000 ns/op
BenchmarkWorkerPool10
BenchmarkWorkerPool10-12         55 20580624 ns/op
BenchmarkWorkerPool100
BenchmarkWorkerPool100-12        52 19341110 ns/op
BenchmarkWorkerPool1000
BenchmarkWorkerPool1000-12       52 19896781 ns/op
BenchmarkWorkerPoolNumCpu
BenchmarkWorkerPoolNumCpu-12     56 19620621 ns/op
PASS
ok      hw7 7.619s
```

Benchmark result, ran with `go test -bench=. -v`