**Goroutines**

Goroutines are functions that are created and scheduled to be run independently by the Go scheduler. The Go scheduler is responsible for the management and execution of goroutines.

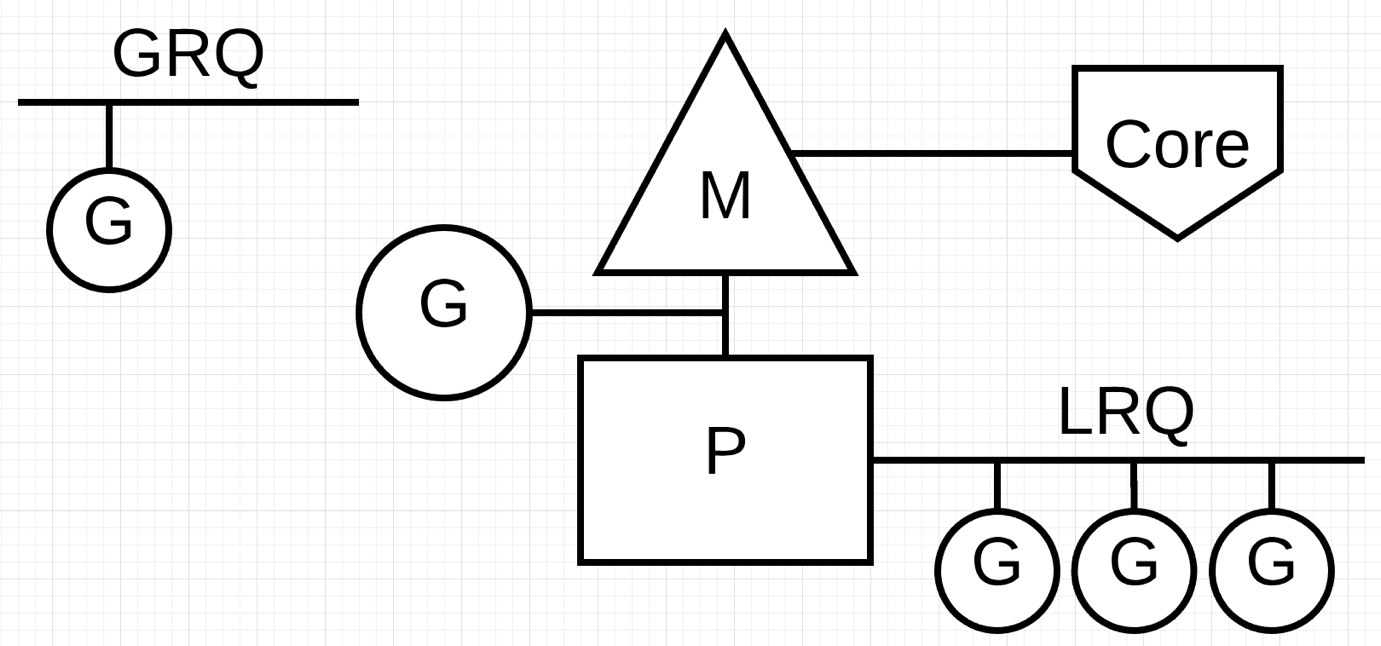
**Notes**

* Goroutines are functions that are scheduled to run independently.
* We must always maintain an account of running goroutines and shutdown cleanly.
* Concurrency is not parallelism.
  + Concurrency is about dealing with lots of things at once.
  + Parallelism is about doing lots of things at once.

*"Parallelism is about physically doing two or more things at the same time. Concurrency is about undefined, out of order, execution." - William Kennedy*

*"By default, goroutines shouldn't outlive the function they were created from. this forces you into a extremely good design posture." - Peter Bourgon*

**How the scheduler works.**

[](https://github.com/ardanlabs/gotraining/blob/master/topics/go/concurrency/goroutines/scheduler.png?v=2)

[Scheduling In Go - Part I](https://www.ardanlabs.com/blog/2018/08/scheduling-in-go-part1.html)    
[Scheduling In Go - Part II](https://www.ardanlabs.com/blog/2018/08/scheduling-in-go-part2.html)    
[Scheduler Tracing In Go](https://www.ardanlabs.com/blog/2015/02/scheduler-tracing-in-go.html)    
[Concurrency, Goroutines and GOMAXPROCS](https://www.ardanlabs.com/blog/2014/01/concurrency-goroutines-and-gomaxprocs.html) 