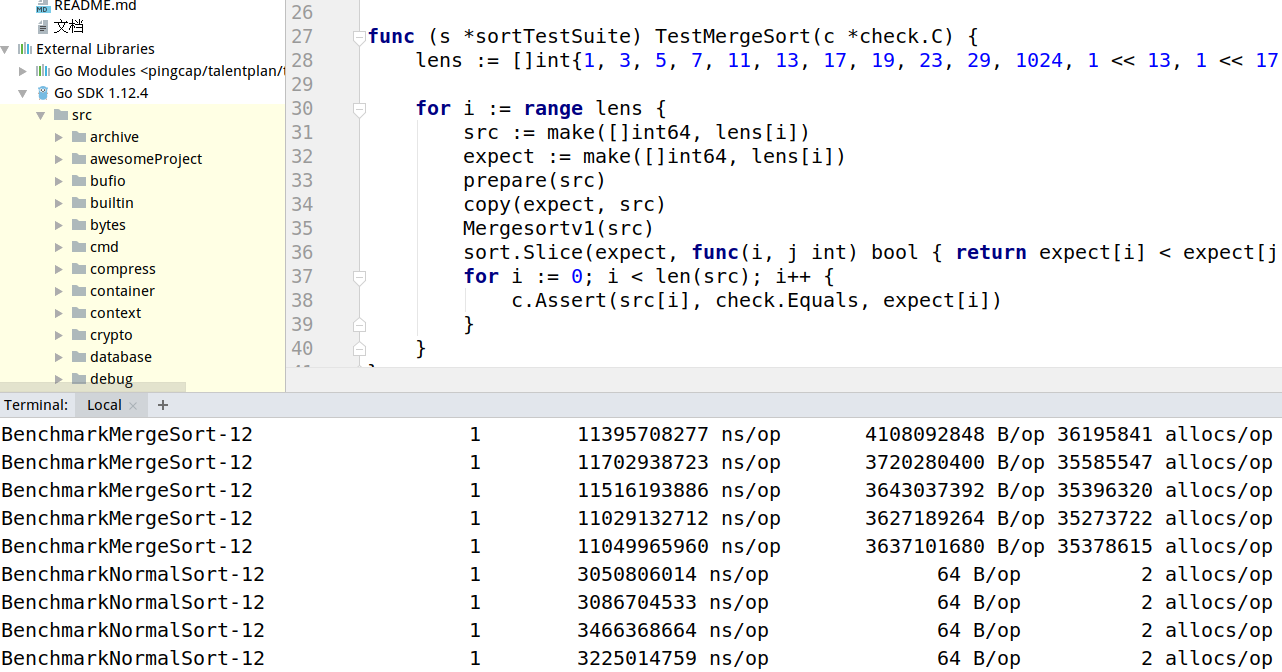
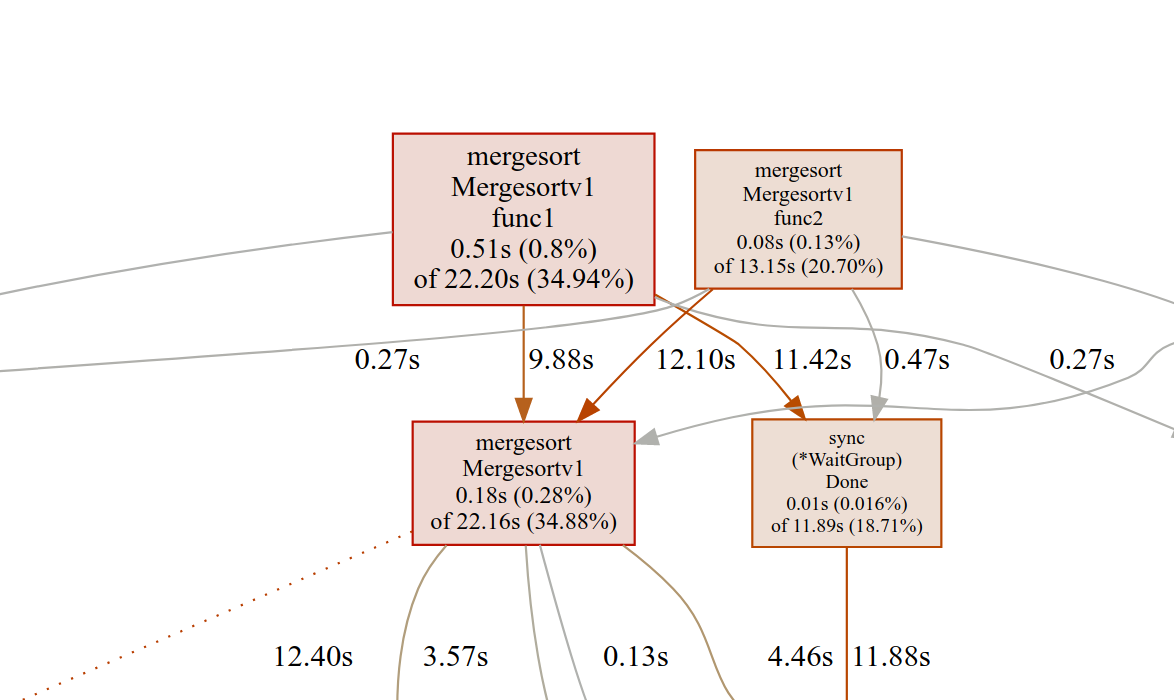


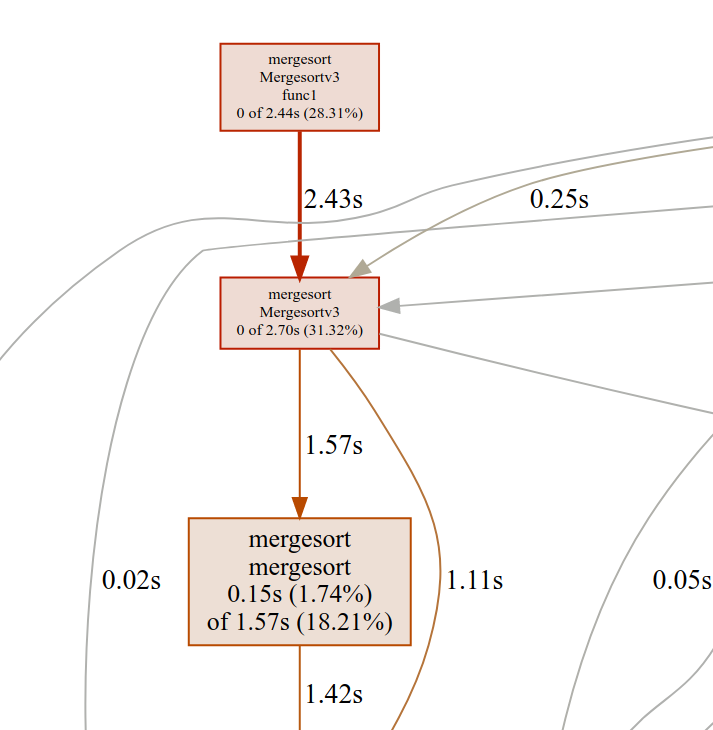
根据pprof图判断性能瓶颈是是在merge和mergesort，这两处时间开销很大，我选择在mergesort这里做优化，当写完简单并发mergesort之后发现性能并没有得到提升，反而因为大量的goroutine开销导致时间是一次排序的三倍。（如下图所示）。



所以在并发mergesort的情况性能瓶颈在goroutine的开销上，我们应该思考如何减少goroutine的开销，假如我有一个512个元素的切片，那么作用于mergesort的就有256，以此类推直到1个元素，但此时我们还要为它单独启一个goroutine,这就很浪费了，所以我们设置一个最大值max在小于max时使用循序实现（mergesort），在大于max时使用并发实现(v1)。



(v1)



(v3)

到此性能已经提升很多。