序

第一次与大家见面，如果看到有问题的地方，欢迎指正。同时也欢迎联系我，或者留言评论，发布论坛。不让错误的东西存在时间太长。

写程序过一段时间后发现有些内容还是容易忘记。有时会记录一些，有时便会忘记记录，过段时间再遇到相同的问题会很难受。要是基础问题，那就是丢人，脸疼了。因此本书是针对基础的工具书。

工作过小几年的朋友应该都清楚，再次遇到面试官。他们依然是要求基础要过硬，其它的都好说。基础不够硬，会让人家很为难的。老艾也是经常去网上各种搜索，又是看各种版本的书，去扫除脑袋里问号。有很多书词汇过于专业，咬文嚼字的过于多。有的更是非程序人员翻译的国外书，那看起来酸爽。

其中《GO语言圣经》是翻译的不错的基础类书籍。专业词翻译的也都非常好。Go现在的好书越来越多，而且大多自己创作，可谓是匠心著作。希望我也能是你们中的一员吧。

本书本着由浅入深的思路。所以新手一般看章节的前半部分即可，其它的回头再看。本书从定义，常规，特性，性能，内存，运行时太，以及一些Go隐形问题，以及建议所构成。

**Go简书**

创始人[Robert Griesemer](http://research.google.com/pubs/author96.html" \t "http://books.studygolang.com/gopl-zh/_blank)，[Rob Pike](http://genius.cat-v.org/rob-pike/" \t "http://books.studygolang.com/gopl-zh/_blank)和[Ken Thompson](http://genius.cat-v.org/ken-thompson/" \t "http://books.studygolang.com/gopl-zh/_blank)三位大牛。他们就职于Google公司，于2007年9月开始设计和实现，然后于2009年的11月对外正式发布。

GO是一个开源的编程语言。有时会被喜欢的它的人称“21世纪的C”。相比于C和C++它易于构建简单、可靠和高效的软件。

Go的编译速度很快。不用在这点担心它。

Go的可移植性很高。不用担心在什么平台下写的。

Go的文化“简单哲学”。

Go的生态已成熟，标准库和第三方的库越来越多。

Go具有CSP并行特性，非常适合服务端编程。

Go有类C的语句，但它的来源先祖不仅仅是C。

Go是纯粹的函数式语言。

Go有GC内存垃圾自动回收。GC运行微秒级。

Go的构建工具go tool越来越完整。

Go少了面向对象类语言很多部件，像宏，异常，继承，运算符重载，构造析构函数等等它都没有要求在语言里。

**本书结构**

第一章是为了对go有兴趣的人而作，即使你没有太多其它语言基础，也可以轻松的运行第一个程序，从而敲响golang的大门。