

《Go 并发编程实战》勘误 v20150501 欲获知最新动态请访问： http://www.ituring.com.cn/book/1525		
页码	勘误内容	修正印次
4	第二段第二行：“与 PHP 一样，Python 的 Ruby 是动态类型的解释型语言”改为“与 PHP 一样，Python 和 Ruby 也是动态类型的解释型语言”	2
7	第 12 行“默认情况下，该目录会包含 go、godoc 和 gofmt 这 3 个工具。本书将在 2.3 节中给出这 3 个工具的简单介绍，并给出免费的 Go 命令教程的网址”改为“默认情况下，该目录会包含 go 和 gofmt 这两个工具。本书将在 2.3 节中给出这些工具的简单介绍，并给出免费的 Go 命令教程的网址”	2
10	第三段的开头有“Go 代码必须放在工作区中。”应改为“一般情况下，Go 源码文件都需要放在工作区中。但是对于命令源码文件来说，这不是必须的。”	3
14	第 14 页的示例错误： <pre>import basic/set import helper/ds import cnet/ctcp import logging</pre> 应为： <pre>import "basic/set" import "helper/ds" import "cnet/ctcp" import "logging"</pre> 加半角引号。	2
16	第 3 段第 3 行有“并且，当前代码包中的所有全局变量的初始化会在代码包初始化函数执行前完成。”应改为“并且，对于每一个代码包来说，其中的所有全局变量的初始化都会在该代码包的初始化函数执行前完成。”	3
17	应该把后面这句话追加到第 4 段之后并自成一段：“此外，被导入的代码包的初始化函数总是会在导入它的那个代码包的初始化函数之前执行。”	3
17	在 2.3 节“doc：用于执行 godoc 命令以打印指定代码包。”，由于在 go 1.3 中实际上已经废除了这个子命令，所以建议把这句话删掉。 为了配合上述修改，还需要修改第 18 页中的一句话：“Go 语言还自带了很多有用的命令和工具。比如上面提到的 godoc 命令和 gofmt 命令。”这句话改为：“Go 语言还附带了很多有用的命令和工具。比如上面提到的 gofmt 命令，以及用于展示指定代码包的文档的 godoc 命令（需要另行安装）。”	2
17	第 8 行，被赋予类型 string 的零值——“”，中文的双引号“”改为英文双引号""。	2
25	倒数第 3 段有“它可以看作是 uint32 类型的一个别名类型，其潜在类型就是 uint32。”，“uint32”应改为“int32”	3
31	第 6 行到第 4 节地址操作符之间的示例存在错误，示例中的逻辑与运算改为逻辑或运算。正文描述进行相应的修改。改后的示例分别如下所示： <pre>b1 b2 !b3 b4 ((b1 b2) !b3) b4 b1 (b2 !b3) b4</pre>	2
34	第 3 段倒数第 2 行“前提是这个变量 v 的值不能是 nil”应改为“前提是这个变量 x 的值不能是 nil”，即“v”应为“x”	2
44	表 3-4 中的第 3 行的说明那一列有“无符号整数类型”应改为“有符号整数类型”	3
45	第 1 段的第 3 行有“而类型 rune 可以被看作是 uint32 的别名类型”应改为“而类型 rune 可以被看作是 int32 的别名类型”	3
45	中间表格说明有符号的 int64 范围时，最后一行的最左边数字的最前面多了一个 2	3
52	以“注意，”开始的那一段删除“如果变量 array1 的值为 nil，那么索引表达式在被求值时就会引发一个运行时恐慌。而”	2
52	删除倒数第 1 段中的“同时，切片的长度是可变的。”	3
59	中间以“显然，”开头的那段最后的“索引值最大的那两个元素值”应改为“索引值最大的那四个元素值”	3
61	“3.2.4 字典”小节的第 3 行“在一个键集对集合中”应改为“在一个键值对集合中”	2
61	倒数第 5 行，“比较操作符=”应为“比较操作符==”。	2
93	第 6 段“x[3]在被求值的时候”改为“x[3]在被求值的时候”	2
102	第 3 个代码片段： <pre>go []string []string</pre> 应该把 go 去掉，应为： <pre>[]string []string</pre>	2

104	倒数第 5 段的倒数第 2 行有“只有具有可比性的值才可以被作为比较操作符==和!=的操作数。”应改为“只有具有可比性的值才可以被作为比较操作符==和!=的操作数。”	3
107	以“一个例外情况是”开始的那一段的最后一行“如果有一个结构体类型是这样的”应改为“如果有一个自定义类型是这样的”	3
127	第 2 段的第 6 行有“而 rune 类型则是 uint32 的别名类型”应改为“而 rune 类型则是 int32 的别名类型”	3
161	示例代码中有两处错误: one.Len() 应该改为 set.Len() ; one.Contains(v) 应该改为 set.Contains(v)	2
189	第 17 行“我们为什么要在这里特别强调样本测试函数的命名呢?你会在 5.3 节中找到答案。”这里的 5.3 节应该改为 5.2 节。	2
208	第 6 章的首页, 倒数第三段“不过那时并没有任何理论基础来支持这种编程方式”改为“不过那时并没有任何理论基础来支持这种编程方式”	2
227	第二段末尾“会在其中一端为就绪前阻塞”改为“会在其中一端未就绪前阻塞”	3
227	第四段第二行的开始处“第二个结果只是代表了”改为“第二个结果值是代表了”	3
241	图 6-11 中第三行的左边两个文本框中的文字(即“TCP”和“UDP”)应该互换。	3
245	跨 245 页和 246 页的那一小段文字(从“注意,”到“两倍。”)所描述的情况已与 Go 语言的当前实现不符, 因此需要删除。	3
261	第一段最后的“标准准化”应改为“标准化”	3
262	“2. 线程间的控制”的“终止线程”的“终止线程”那一段中“`pthread_cancel 函数的作用是” 改为“pthread_cancel 函数的作用是”	2
289	“Go 语言有能力祝你一臂之力。”应改为“Go 语言有能力助你一臂之力。”	2
292	图 6-31 下面的那段中的“字段 mstartfm”应该是“字段 mstarfn”	3
295	最后一段中“每一个 P 中都一个可运行 G 队列”应改为“每一个 P 中都有一个可运行 G 队列”	3
301	301 页, 图 6-34 左边的第二个标注的文字“是否先检查调度器队列的可运行 G 队列”改为“是否先检查调度器的可运行 G 队列”	3
302	中间以“只要”开头的那段中的“倘若从队列中不到可运行的 G”应改为“倘若从队列中找不到可运行的 G”	3
306	图 6-36 中左上角的那个菱形线框的左边箭头连接线的下面应该有个“是”字, 而其下面的箭头连接线的左边应该有个“否”字	3
312	以“我们把”开头的那段的第一句中的“schedtace.go”应改为“schedtrace.go”	3
317	最后一段中的“至于这个新的可运行 G 什么会被执行”应改为“至于这个新的可运行 G 什么时候会被执行”	3
337	第一段示例代码中的“Handle(orig Person)”应改为“Handle(orig *Person)” 第二段示例代码中的“Handle(orig Person)”应改为“Handle(orig *Person)”	3
338	第一段示例代码中的“handler.Handle(p)”应改为“handler.Handle(&p)” 第二段示例代码中的“handler.Handle(p)”应改为“handler.Handle(&p)”	3
343	第二段示例代码中的“handler.Handle(p)”应改为“handler.Handle(&p)”	3
349	第四段的倒数第二行“作用域更大的那个 ok 变量的值会一直是 false”改为“作用域更大的那个 ok 变量的值会一直是 true”	3
349	倒数第二段“针对通道的接收操作(以及发送操作)并不没有超时这一概念” 改为“针对通道的接收操作(以及发送操作)并没有超时这一概念”, 即去掉多余的“不”字。	2
355	倒数第三段倒数第四行“.....就会被阻塞住” 应该为“.....就会被阻塞住”	2
391	第二块示例代码的第 6、7 行的“int(rand.Int31n(1000))”均需改为“int(rand.Int31n(1000) + 1)”	3
392	第二段的最后有“在这里, 这个范围是[0,1000)。”，需改为“在这里, 这个范围是[1,1001)”	3
404	第 3 段第 2 行末尾“就会已发一个运行时恐慌” 改为“就会引发一个运行时恐慌”	2
410	df.fmutex.Lock() df.fmutex.Unlock() _, err = df.f.Write(bytes) 应该是: df.fmutex.Lock() defer df.fmutex.Unlock() _, err = df.f.Write(bytes) 中间那行的最左边少了“defer ” (“defer”和一个空格)。	2
412	第二段: “操作地行为和意义”应为“操作的行为和意义”	2
413	以“一定不要忘记”开头的那段的最后一行“我们应该调用 fmutex 字段的 RLock 方法”改为“我们应该调用 fmutex 字段的 RUnlock 方法”	3
417	第 3 段第 1 行“在创建互斥量和不形成临界区的情况下”改为“在不创建互斥量和不形成临界区的情况下”	2
425	“对一个 sync.WaitGroup 类型值的 Add 方法的的第一次调用”改为“对一个 sync.WaitGroup 类型值的 Add 方法的第一次调用”	2

434	第 13 行的代码“b.Logf("The length of % value is %d.\n", mapType, m1)”改为 “b.Logf("The length of %s value is %d.\n", mapType, m1)”	3
450	第 450、455 页中的“IdGenertor”都修正为“IdGenerator”	
461	第 461、462、464、465 页，示例中的 Receive a element 应改为 Received an element	2
461	第 461、465 页，示例和文字描述中的 Closed chan1 应改为 chan1 closed	2
461	第 461、462、463 页，示例中的 Receive element from chan1 应改为 Begin to receive elements from chan1	2
480	第 480、481、482、483 页中的“IdGenertor”都修正为“IdGenerator”	
504	第二段示例代码，去掉重复的“if err != nil {”和“}”	3
557	第 2 项中的“http://gogits.org”应改为“http://gogs.io”	3