

Go Programming Fundamentals

Assignment #5

Due Date:

Race Condition and Golang Channels

Programming Requirements

- Submit your assignment into a **single Go language file**.
- You must write **your name** at the top of your assignment source list.
- Write your **compiler version and operating system name** at the top of your assignment source list.
- Assignment that is turned in late will lose **one point per day** starting after the due date.
- Make sure that you do appropriate **error checking** in your program. (User-friendliness)
- Do not turn in **incomplete or crashing program**, you will receive **zero points**.
- Make sure to read **grading policy** carefully that will tell you how your assignment is graded.

Assignment #5 Grading Policy

Category	Points Possible	Points Received
Correctness and Efficiency .	10	
Meaningful variable names	10	
Fix race condition.	25	
Use golang channels	35	
Complete Documentation	10	
User-friendliness and Code Readability	10	
Total	100	

Go Programming Fundamentals

Assignment Description

Part A

Following program has a race condition. Fix the race condition in this code. Now rewrite this code using golang channels. To see the race condition run this program using -race switch.

```
1 package main
2
3 import (
4     "fmt"
5     "sync"
6 )
7
8 func main() {
9     var wg sync.WaitGroup
10    wg.Add(5)
11    for i := 0; i < 5; i++ {
12        go func() {
13            fmt.Println(i) // Not the 'i' you are looking for.
14            wg.Done()
15        }()
16    }
17    wg.Wait()
18 }
19
20
```

Go Programming Fundamentals

Assignment Description

Part B

Following program has a race condition. Fix the race condition in this code. Rewrite this code in a separate file and use golang channels so there is no race condition. To see the race condition run this program using -race switch.

```
1 package main
2
3 import (
4     "fmt"
5     "sync"
6 )
7
8 var global int
9 var wg sync.WaitGroup
10
11 func count() {
12     defer wg.Done()
13     for i := 0; i < 10000; i++ {
14         global++
15     }
16 }
17
18 func main() {
19     wg.Add(2)
20     go count()
21     go count()
22     wg.Wait()
23     fmt.Println(global)
24 }
```