**Data race conditions**

Data race happens when we have two or more goroutines accessing the same variable concurrently and at least one of the accesses is a write access.

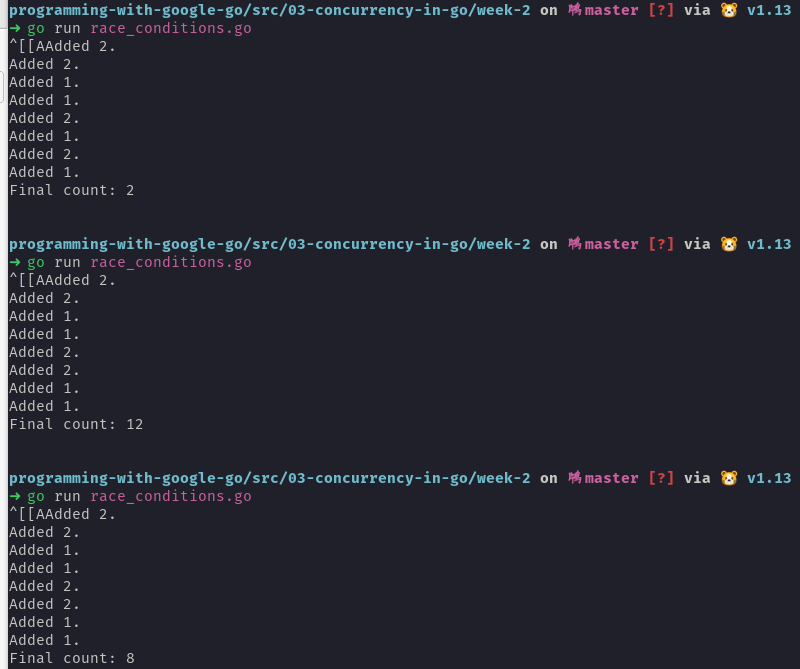
Once the execution order is non-deterministic, we can have different results at every execution.

In the exercise, I created two functions, **incrementOne** and **IncrementTwo**, to make easier to see the different results of the race conditions.

The main function runs twice, calling each of the "increment" functions concurrently.

In this case, as we can't determine the execution order, we can have various different results between 2 and 12.

By printing the value the functions are adding at every execution, we can observe the execution order of each time we call “go run race\_conditions.go”, as we can see in the picture bellow:



We could see the race happening using only one of the "increment" functions and changing the **wait.Add(2)** to **wait.Add(1)**, in line 14.