Problem 5: Barbershop

This is another problem originally proposed by Dijkstra. There is a barbershop with a number of chairs for waiting. If there are no customers the barber goes to sleep. If a customer enters and there are no chairs available to wait, the customer leaves. There is one barber chair. If a customer comes in and the barber is asleep, the customer wakes up the barber. This problem emulates inter process communication and synchronization between multiple operating system processes. There is no point in a host process being awake if there are no processes waiting to being served. There is also only enough room for a certain number of processes to queue and wait for a turn.

Approach 1 (BS_go): Go Semaphores implementation: source: github.com/soniakeys/LittleBookOfSemaphores

Approach 2 (BS_m2): Go Mutex 2 implementation: source: github.com/soniakeys/LittleBookOfSemaphores

Barbershop	**Approach 1: Go Semaphores**	**Approach 2: Go Mutex 2**
Correctness	Output gets jumbled sometimes and occasionally looks incoherent. With running 4000 waiting chairs and 6000 customers, the behaviour of this implementation seems mostly fine. Occasionally there are numerous customers arriving quickly at the same time and leaving because all of the chairs are full. This happens when the barber is sleeping although he eventually does wake up.	This implementation seems more correct than Approach 1. The barber stays busy and there aren't large numbers of customers leaving because the waiting area is full. There is no garbled output or other errors that are easily discernible. In a way, the output from this implementation almost looks like it was programmed without threads as it churns along almost perfectly except for the odd thread being out of order.
Comprehensibility	This approach uses Sonia Keys semaphore implementation and is almost exactly like the solution presented in The Little Book of Semaphores. A WaitGroup is used to ensure that all of the GoRoutines have a chance to finish. This implementation is more comprehensible given that is follows a familiar pattern. This solution also uses only a few lines of code.	Approach 2 is a non-semaphore version that uses mutexes and channels to communicate. An unbuffered channel called barberRoom is used so that a customer can send his customer number to the Barber. Another channel called cutDone is used so that the Barber can send a message saying that the haircut has finished. After getting used to the way Channels work this solution becomes very easy to read.
Performance	4 chairs/ 6 customers Avg. 0.27 ms. 40 chairs/ 60 customers Avg. 1.8 ms. 4000 chairs/ 6000 customers Avg. 165 ms.	4 chairs/ 6 customers Avg. 5.1 ms. 40 chairs/ 60 customers Avg. 49 ms. 4000 chairs/ 6000 customers Avg. 5000 ms.

Barbershop	Approach 1: Go Semaphores	Approach 2: Go Mutex 2
Output: 4 haircuts/ 6 customers	barber sleeping customer 1 arrives, 0 customers customer 1 waits barber cutting hair customer 1 gets hair cut barber sleeping customer 6 arrives, 1 customers customer 6 waits barber cutting hair customer 6 gets hair cut barber sleeping customer 2 arrives, 2 customers customer 2 waits barber cutting hair customer 2 gets hair cut barber sleeping customer 3 arrives, 3 customers customer 3 waits barber cutting hair customer 3 gets hair cut barber sleeping customer 4 arrives, 4 customers customer 4 shop full, leaves customer 5 arrives, 4 customers customer 5 shop full, leaves customer 1 leaves with hair cut customer 6 leaves with hair cut customer 2 leaves with hair cut customer 3 leaves with hair cut 268.312µs	barber sleeping customer 2 arrives, 0 customers customer 2 waits customer 6 arrives, 1 customers barber wakes, cuts customer 2 hair customer 6 waits customer 3 arrives, 2 customers customer 3 waits customer 4 arrives, 3 customers customer 4 waits customer 5 arrives, 4 customers customer 5 ,shop full, leaves customer 1 arrives, 4 customers customer 1 ,shop full, leaves customer 2 getting hair cut customer 2 leaves with hair cut barber sleeping barber wakes, cuts customer 6 hair customer 6 getting hair cut customer 6 leaves with hair cut barber sleeping barber wakes, cuts customer 3 hair customer 3 getting hair cut customer 3 leaves with hair cut barber sleeping barber wakes, cuts customer 4 hair customer 4 getting hair cut customer 4 leaves with hair cut barber sleeping 5.457417ms