

Especialización en Back End III

Mesa de trabajo

- Ejercitación en grupo
- Nivel de complejidad: medio 🔥 🔥

Objetivo

El objetivo de esta guía práctica es que podamos afianzar los conceptos sobre concurrencia en Go, usando goroutines y channels.

Consigna

Se requiere un programa para calcular si un número es par o impar desde una lista de **int**. Para esto se deben crear dos funciones:

- **par(chan int)** debe recibir a través de un channel de int los valores pares enviados y mostrarlos por la terminal.
- **impar(chan int)** debe recibir a través de un channel los valores impares y mostrarlos por terminal.

El programa debe comenzar dos goroutines que se ejecuten concurrentemente. En la goroutine principal se itera el slice provisto con valores y, en caso de que el valor sea par, se lo debe enviar al channel correspondiente. De igual manera para los impares.

Ayuda: Si el resto de una división por dos da cero es un número par (`i%2 == 0`) y si da 1 es impar (`i%2 == 1`). Para obtener el resto de una operación, utilizar el operador %: `i%2 == 1`.

¡Éxitos!