




PRÁCTICA 2 – DICIEMBRE 2020


SISTEMAS DISTRIBUIDOS.


GRADO EN INGENIERÍA TELEMÁTICA – ETSIT, URJC

 **Profesor:** Felipe Ortega.

 **Versión:** 1.0.

 **Lenguaje de programación:** Go.

 **Fecha tope de entrega:** 22:00 del lunes, 11 de enero de 2020.

 **Envío:** Código fuente **que incluya comentarios**, a través del espacio de entrega en Aula Virtual.

1 OBJETIVOS

- Comprender las estrategias y primitivas de sincronización entre gorutinas.
- Comprender los problemas de concurrencia asociados al uso de *goroutines*.
- Comprender la utilización de *channels* para comunicación y sincronización entre *goroutines*.
- Consideración de patrones para diseño de programas concurrentes.

EL PROBLEMA DE SANTA CLAUS

Este problema está basado en el problema homónimo de ¹ (cf. sec. 2.3.7).

¹ David Vallejo, C. G. and Jiménez, J. A. A. (2016). *Programación concurrente y tiempo real*. CreateSpace, 3 edition

2 ENUNCIADO DEL PROBLEMA

Próximamente ya las fechas de Navidad, Santa Claus está en su casa cerca del Polo Norte. Para poder despertarlo de su descanso, se debe cumplir una de las dos condiciones siguientes:

- Que los 9 renos disponibles para tirar del trineo hayan regresado de vacaciones.

- Que alguno de los elfos ayudantes necesiten que le eche una mano para fabricar un juguete.

Para intentar evitar importunar al jefe más de lo debido, los elfos han acordado despertarle si 3 de ellos tienen problemas. En el caso de que un grupo de 3 duendes ya le haya despertado, el resto de duendes con problemas deben esperar a que Santa Claus acabe de ayudar al primer equipo.

Sin embargo, en caso de que en algún momento todos los renos vuelvan de vacaciones, Santa Claus acaba de atender a los elfos a los que estaba ayudando y deja a los demás en espera y prepara el trineo. El reparto es más importante que ayudar a fabricar juguetes, que se pueden entregar el año siguiente. El último reno en llegar es el encargado de avisar a Santa Claus, mientras que cuando todos los anteriores renos van llegando se quedan en espera, antes de ser enganchados al trineo.

2.1 CONDICIONES ADICIONALES

Para concretar la resolución del problema, se establecen las siguientes condiciones adicionales:

- En la fábrica de juguetes hay $D = 12$ elfos trabajando.
- Los renos irán llegando aleatoriamente, con un intervalo de, al menos, 5 segundos entre la llegada de dos renos consecutivos.
- Cada elfo fabrica un juguete. Uno de cada tres juguetes da problemas.
- Santa Claus invierte un tiempo aleatorio de entre 2 y 5 segundos en ayudar a un equipo de 3 elfos con problemas.
- En el caso de que Santa Claus ayude a todos los elfos y todavía no hayan llegado todos los renos del trineo, se vuelve a dormir hasta que le vuelvan a despertar.

3 PLANTEAMIENTO

Además de la función principal, se recomienda crear, al menos, 3 funciones para gorutinas específicas:

- Una gorutina que modela a Santa Claus.
- Una gorutina que modela el comportamiento de un duende.
- Una gorutina que modela el comportamiento de un reno.

Respecto a los eventos que deben ser sincronizados, será preciso tener elementos para despertar a Santa Claus, notificar a los renos que se han enganchado al trineo y controlar la espera de los elfos cuando otro grupo de 3 elfos esté siendo ayudado por Santa Claus.

4 EVALUACIÓN

Se deberá entregar un esquema y planteamiento teórico, donde se describa el problema así como las estrategias y mecanismos de sincronización que se han puesto en práctica para resolver todos los aspectos operativos solicitados en el enunciado.

Además, se entregará el código fuente en lenguaje de programación Go, comentado, implementando el esquema propuesto.

- Corrección del planteamiento teórico para resolver el problema → **3 puntos**.
- Corrección funcional del programa → **4.5 puntos**.
- Corrección de estructura del programa y estilo → **2.5 puntos**.

REFERENCIAS

David Vallejo, C. G. and Jiménez, J. A. A. (2016). *Programación concurrente y tiempo real*. CreateSpace, 3 edition.

