

Semáforos - Santa Claus y los Renos

Rodrigo García Carmona



Este problema demuestra el uso de semáforos para coordinar tres tipos de procesos (Santa Claus, renos y elfos).

Santa Claus duerme en su tienda en el Polo Norte y sólo puede ser despertado porque:

1. Todos los renos han vuelto de sus vacaciones en el Pacífico.
2. Algunos elfos tienen problemas fabricando juguetes y deben preguntar dudas a Santa Claus.

Para que Santa pueda dormir, los elfos sólo pueden despertarle cuando al menos tres de ellos tengan problemas. Si tres elfos están haciendo preguntas para solucionar sus problemas, cualquier otro elfo que desee preguntar a Santa Claus deberá esperar a que esos 3 elfos vuelvan.

El último reno en volver del Pacífico debe buscar a Santa Claus mientras los otros esperan en un establo calentito. Sólo cuando estén de vuelta todos los renos Santa les enganchará a su trineo.

Si Santa Claus se despierta y encuentra tanto a tres elfos esperando como al último reno que tenía que volver del Pacífico, Santa Claus tiene decidido que los elfos pueden esperar hasta después de Navidad, porque es más importante tener su trineo listo y empezar a repartir regalos (se asume que los renos no desean abandonar los trópicos y que por tanto están allí hasta el último momento posible).

Resuelva este problema usando semáforos para 10 renos.

Solución

```
#define REINDEER 9    /* máximo num. de renos que se pueden estabular */
#define ELVES 3      /* tamaño del grupo de elfos */

/* Semáforos */
only_elves = 3,      /* 3 pueden ir con Santa */
emutex = 1,         /* mutex para elf_cnt */
rmutex = 1,         /* mutex para rein_ct */
rein_wait = 0,      /* establo para que los renos esperen */
sleigh = 0,         /* los renos esperan junto al trineo */
done = 0,           /* juguetes entregados */
santa_signal = 0,   /* hasta 2 elfos esperan fuera de la casa de Santa */
santa = 0,          /* Santa duerme en este semáforo */
problem = 0,        /* espera para hacer preguntas a Santa */
elf_done = 0;       /* han recibido respuesta de Santa */

/* Variables compartidas */
rein_ct = 0;        /* contador de renos que han vuelto */
elf_ct = 0;         /* contador de elfos con problemas */
```

```

/* Proceso Reno */
for (;;) {
    ...
    en la playa del Pacífico hasta que se acerca la Navidad
    ...
    wait (rmutex)
    rein_ct++
    if (rein_ct == REINDEER) { /* si es el último reno */
        signal (rmutex)
        signal (santa) /* despierta a Santa */
    }
    else {
        signal (rmutex)
        wait (rein_wait)
    }
    /* todos los renos esperan junto al trineo */

    wait (sleigh)
    ...
    entregando regalos
    ...
    wait (done)
    ...
    de vuelta a la playa del Pacífico
    ...
}

/* Proceso Elfo */
for (;;) {
    wait (only_elves) /* sólo entran 3 elfos a la vez */
    wait (emutex)
    elf_ct++
    if (elf_ct == ELVES) { /* si es el tercer elfo */
        signal (emutex)
        signal (santa) /* despierta a Santa */
    }
    else {
        signal (emutex)
        wait (santa _signal) /* esperan en la puerta */
    }
    wait (problem)
    ...
    haciendo preguntas /* Santa despierta al elfo */
    ...
    wait (elf_done)
    signal (only_elves)
}

/* Proceso Santa */
for (;;) {
    wait (santa) /* Santa duerme */

```

```

wait (rmutex)      /* Mutex para rein_ct */
if (rein_ct == REINDEER) { /* ha llegado el último reno */
    rein_ct = 0      /* reseteo mientras está bloqueado */
    signal (rmutex) /* Mutex para rein_ct */
    for (i = 0; i < REINDEER - 1; i++)
        signal (rein_wait)
    for (i = 0; i < REINDEER; i++)
        signal (sleigh)
    ...
    entregando juguetes y volviendo
    ...
    for (i = 0; i < REINDEER; i++)
        signal (done)
}
else { /* han llegado 3 elfos */
    signal (rmutex) /* Mutex para rein_ct */
    for (i = 0; i < ELVES - 1; i++)
        signal (santa_signal)

    wait (emutex) /* Mutex para elf_ct */
    elf_ct = 0
    signal (emutex) /* Mutex para elf_ct */
    for (i = 0; i < ELVES; i++) {
        signal (problem)
        ...
        respondiendo a la pregunta
        ...
        signal (elf_done)
    }
}
}
}

```