Sistemas Distribuidos y Verificación
Tarea 8

Fabián Romero Jiménez

Problema 1 En la fábula entre productores y consumidores, se asumió que Bob puede ver si la lata en la ventana de Alice esta arriba o abajo. Diseñar un protocolo utilizando latas y cuerdas que funciona si Bob no puede ver el estado de las latas. Demuestra que el protocolo cumple safety y liveness.

Solución Se pone una lata en la ventana de cada lado, cada quién puede tirar a la opuesta y levantar la suya.

Protocolo de Alicia.

A1 Si los perros tienen hambre, revisa si no hay comida, en ese caso levanta su lata (si está tirada) y entonces tira de la lata de Bob.

A2 Espera a que su lata este caida. Entonces saca a los perros, cuando estos regresan, realiza la Acción A1.

Protocolo de Bob.

B1 Espera a que su lata esté caida, entonces sale al patio y pone comida, cuando regresa, levanta su lata y tira la lata de Alicia.

Propiedades buscadas.

(a) Safety - Exclusion mutua. Bob y los perros no están nunca en el patio juntos.

(b) No starvation. Si Bob siempre esta listo para poner comida y los perros siempre tienen hambre, entonces los perros comerán muy frecuentemente.

(c) Productor-Consumidor. Los perros no entraran al patio a menos que haya comida y Bob no pondrá comida si no se ha agotado.

Veamos que siempre se cumplen las propiedades.

Exclusión mutua
Observese que se puede intrepretar lata de Bob abajo: Ya no hay comida

y que los perros no están en el patio. lata de Alice abajo: Ya hay comida y Bob no está en el patio.

Caso 11 (Ambas latas arriba), así empieza el protocolo. A1) Al final de este paso, si no hay comida establece la latas a 10, En todo este tiempo, ni los perros ni Bob salen al patio. A2) No se ejecuta, pues solo lo hace si la lata de Alice esta abajo B1) No se ejecuta, pues solo lo hace si la lata de Bob esta abajo

Caso 10 (Alice arriba, Bob abajo). A1) Al final de este paso, si no hay comida establece la latas a 10 (este mismo caso). En todo este tiempo, ni los perros ni Bob salen al patio. A2) No se ejecuta, pues solo lo hace si la lata de Alice esta abajo B1) Bob pone comida, regresa y establece (01) y como la lata de Alice esta todo el tiempo arriba hasta que Bob regresa, los perros no salen.

Caso 01 (Alice abajo, Bob arriba). A1) Al final de este paso, si no hay comida establece la latas a 10. A2) Alice saca los perros, y cuando regresan opera A1. Como la lata de Bob siempre esta arriba y solo se tira cuando los perros se acaban la comida y regresan, no se encuentran en el patio. B1) No se ejecuta, pues solo lo hace si la lata de Bob esta abajo

Caso 00 (Alice abajo, Bob abajo). Esto no puede ocurrir, pues las tres acciones dejan siempre al menos una lata arriba.

Así, la exclusión mutua ya está probada.

No starvation.
Por contradicción, supongamos que no es cierto, es decir: Que llege un momento que los perros tengan hambre, Bob este listo para poner comida pero no lo logre.

Si la lata de Bob estuviera abajo, Bob saldría a poner comida contradidiendo la hipotesis de su disponibilidad. Si la lata de Bob estuviera arriba, Alicia realiza A1 pues los perros tienen hambre y al final tira la lata de Bob, llegando al caso anterior.

Productor-Consumidor.

La propiedad de exclusión mutua implica que los perros y Bob nunca será en el patio juntos. Bob no entrará en el patio hasta que Alice tire la lata de Bob, que ella va a hacer sólo si no hay más comida. Del mismo modo, los perros no entraran en el patio hasta que Bob tire la lata de Alicia que no sucedera sino hasta que Bob haya puesto comida y regresado a casa.

Problema 2 Suponiendo que se tiene un programa 30% serial y 70% paralelizable. si queremos hacerlo n veces más rápido, cuantos núcleos de proceso serán necesarios.

Respuesta Usando la ley de Amdahl que nos da el speedout. $speedout = \frac{1}{(1-p)+\frac{p}{n}}$
o despejando la n se tiene $n = \frac{p}{\frac{1}{speedout}-(1-p)}$

speedout 1: 1 un núcleo es suficiente pues $\frac{0.7}{\frac{1}{1}-(0.3)} = 1$.

speedout 2: 3.5 núcleos son suficientes $\frac{0.7}{\frac{1}{2}-(0.3)} = \frac{0.7}{0.2} = 3.5$.

speedout 3: 21 núcleos son suficiente pues $\frac{0.7}{\frac{1}{3}-(0.3)} = \frac{\frac{7}{10}}{\frac{1}{3}-\frac{3}{10}} = \frac{\frac{7}{10}}{\frac{1}{30}} = 21$.

speedout 4 o mayor, ninguna cantidad de nucleos es suficiente, puesto que $speedout = \frac{1}{(1-p)+\frac{p}{n}} < \frac{1}{(1-p)} = \frac{1}{(0.3)} = 1\frac{1}{3} < 3.34$
por lo que al aumentar la cantidad de núcleos el speedout se acercará asintóticamente a $3\frac{1}{3}$ y nunca podrá ser mayor a este número.