

Sincronización entre Procesos II

Matías Barbeito

30/08/2016

Ejercicios **largos** de sincronización entre procesos.

Foco en:

- Cómo encararlos.
- Cómo resolver algunos esquemas típicos de sincronización que aparecen en los enunciados.

La CNRT recibió una denuncia reclamando que muchas líneas de colectivos no recogen a los pasajeros. Ellos sospechan que al no haber suficientes colectivos, éstos se llenan muy rápidamente. Por lo tanto nos pidieron construir un simulador que abarque a los actores e interacciones involucradas.

El simulador debe contar con dos tipos de procesos, colectivo y pasajero. Cada parada está representada por su número en el recorrido. Hay N paradas y M colectivos.

Cada pasajero comienza esperando en una parada (la cual recibe por parámetro) detrás de las personas que ya se encontraban en ella (de haberlas). Una vez que el colectivo llega y el pasajero logra subir, le pide al colectivo que le marque la tarifa con la función `pedirleBoleto()`.

Luego espera a que el colectivo haga `marcarTarifa()` y finalmente el pasajero ejecuta `pagarConSUBE()`. Luego de pagar, procede a `viajar()`. Cuando el pasajero termina de `viajar()`, efectúa `dirigirseAPuertaTrasera()` y

una vez que el colectivo se detiene, los pasajeros que están agrupados en la puerta trasera realizan `bajar()` de a uno por vez, sin importar el orden.

El colectivo recibe como parámetro la capacidad (cantidad de pasajeros) del colectivo, que comienza vacío, y el identificador del colectivo (entre 0 y $M-1$).

Al llegar a una parada, el colectivo, se detiene con `detener()`. Si hay pasajeros esperando para bajar, éste abre su puerta trasera para indicar que ya pueden hacerlo (`abrirPuertaTrasera()`). Mientras esto sucede abre la puerta delantera (`abrirPuertaDelantera()`) y las personas que esperaban en la parada comienzan a ascender en orden siempre y cuando haya capacidad (notar que a esta altura ya se sabe cuántos pasajeros van a bajar).

Las personas proceden a subir y el colectivo los atiende de a uno marcando en la máquina con `marcarTarifa()`, pero nunca lo hace antes de que el anterior haya terminado de `pagarConSUBE()`. Si no hay más pasajeros para subir o se llegó...

al límite de capacidad, el colectivo no duda en `cerrarPuertaDelantera()`, impidiendo que el resto de las personas en la parada ascienda.

Una vez que los pasajeros terminan de ascender, el chofer espera a que terminen de descender todos los pasajeros que así lo desean, procede a `cerrarPuertaTrasera()` y `avanzar()` nuevamente a la siguiente parada, donde la dinámica será la misma.

Puede asumir:

- El colectivo se detiene en todas las paradas y abre la puerta delantera, sin importar si hay pasajeros esperando para bajar o subir.
- El cálculo de la cantidad máxima de pasajeros que el colectivo deja subir puede realizarse considerando la cantidad de pasajeros que ya se sabe que descenderán (no es un problema que brevemente se vea superada la capacidad del colectivo mientras se espera que terminen de descender).

Uffffff

al límite de capacidad, el colectivo no duda en cerrarPuertaDelantera(), impidiendo que el resto de las personas en la parada ascienda.

Una vez que los pasajeros terminan de ascender, el chofer espera a que terminen de descender todos los pasajeros que así lo desean, procede a cerrarPuertaTrasera() y avanzar() nuevamente a la siguiente parada, donde la dinámica será la misma.

Puede asumir:

- El colectivo se detiene en todas las paradas y abre la puerta delantera, sin importar si hay pasajeros esperando para bajar o subir.
- El cálculo de la cantidad máxima de pasajeros que el colectivo deja subir puede realizarse considerando la cantidad de pasajeros que ya se sabe que descenderán (no es un problema que brevemente se vea superada la capacidad del colectivo mientras se espera que terminen de descender).

Uffffff
Arghhhhhhhhhhh

al límite de capacidad, el colectivo no duda en `cerrarPuertaDelantera()`, impidiendo que el resto de las personas en la parada ascienda.

Una vez que los pasajeros terminan de ascender, el chofer espera a que terminen de descender todos los pasajeros que así lo desean, procede a `cerrarPuertaTrasera()` y `avanzar()` nuevamente a la siguiente parada, donde la dinámica será la misma.

Puede asumir:

- El colectivo se detiene en todas las paradas y abre la puerta delantera, sin importar si hay pasajeros esperando para bajar o subir.
- El cálculo de la cantidad máxima de pasajeros que el colectivo deja subir puede realizarse considerando la cantidad de pasajeros que ya se sabe que descenderán (no es un problema que brevemente se vea superada la capacidad del colectivo mientras se espera que terminen de descender).

Uffffff

Arghhhhhhhhhhh

!\$&\$@#!&@!\$&\$

1) Leer detenidamente **todo** el enunciado

La CNRT recibió una denuncia reclamando que muchas líneas de colectivos no recogen a los pasajeros. Ellos sospechan que al no haber suficientes colectivos, éstos se llenan muy rápidamente. Por lo tanto nos pidieron construir un simulador que abarque a los actores e interacciones involucradas.

La CNRT recibió una denuncia reclamando que muchas líneas de colectivos no recogen a los pasajeros. Ellos sospechan que al no haber suficientes colectivos, éstos se llenan muy rápidamente. Por lo tanto nos pidieron construir un simulador que abarque a los actores e interacciones involucradas.

El simulador debe contar con dos tipos de procesos, colectivo y pasajero. Cada parada está representada por su número en el recorrido. Hay N paradas y M colectivos.

La CNRT recibió una denuncia reclamando que muchas líneas de colectivos no recogen a los pasajeros. Ellos sospechan que al no haber suficientes colectivos, éstos se llenan muy rápidamente. Por lo tanto nos pidieron construir un simulador que abarque a los actores e interacciones involucradas.

El simulador debe contar con **dos tipos de procesos**, colectivo y pasajero. Cada parada está representada por su número en el recorrido. Hay **N paradas y M colectivos**.

Cada pasajero comienza esperando en una parada (la cual recibe por parámetro) detrás de las personas que ya se encontraban en ella (de haberlas). Una vez que el colectivo llega y el pasajero logra subir, le pide al colectivero que le marque la tarifa con la función `pedirleBoleto()`. Esta función devuelve el número identificador del colectivo.

Cada pasajero comienza esperando en una parada (la cual recibe por parámetro) **deetrás** de las personas que ya se encontraban en ella (de haberlas). Una vez que el colectivo llega y el pasajero logra subir, le pide al colectivero que le marque la tarifa con la función `pedirleBoleto()`. Esta función devuelve el número identificador del colectivo.

Cada pasajero comienza esperando en una parada (la cual recibe por parámetro) **detrás** de las personas que ya se encontraban en ella (de haberlas). Una vez que el colectivo llega y el pasajero logra subir, le pide al colectivo que le marque la tarifa con la función `pedirleBoleto()`. Esta función devuelve el número identificador del colectivo.

Luego espera a que el colectivo haga `marcarTarifa()` y finalmente el pasajero ejecuta `pagarConSUBE()`. Luego de pagar, procede a `viajar()`. Cuando el pasajero termina de `viajar()`, efectúa `dirigirseAPuertaTrasera()` y una vez que el colectivo se detiene, los pasajeros que están agrupados en la puerta trasera realizan `bajar()` de a uno por vez, sin importar el orden.

Cada pasajero comienza esperando en una parada (la cual recibe por parámetro) **detrás** de las personas que ya se encontraban en ella (de haberlas). Una vez que el colectivo llega y el pasajero logra subir, le pide al colectivo que le marque la tarifa con la función `pedirleBoleto()`. Esta función devuelve el número identificador del colectivo.

Luego **espera** a que el colectivo haga `marcarTarifa()` y finalmente el pasajero ejecuta `pagarConSUBE()`. Luego de pagar, procede a `viajar()`. **Cuando** el pasajero termina de `viajar()`, efectúa `dirigirseAPuertaTrasera()` y una vez que el colectivo se detiene, los pasajeros que están agrupados en la puerta trasera realizan `bajar()` **de a uno por vez, sin importar el orden**.

El colectivo recibe como parámetro la capacidad (cantidad de pasajeros) del colectivo, que comienza vacío, y el identificador del colectivo (entre 0 y $M-1$).

El colectivo **recibe como parámetro la capacidad** (cantidad de pasajeros) del colectivo, que comienza vacío, y **el identificador** del colectivo (entre 0 y $M-1$).

El colectivo **recibe como parámetro la capacidad** (cantidad de pasajeros) del colectivo, que comienza vacío, y **el identificador** del colectivo (entre 0 y $M-1$).

Al llegar a una parada, el colectivo, se detiene con `detener()`. Si hay pasajeros esperando para bajar, éste abre su puerta trasera para indicar que ya pueden hacerlo (`abrirPuertaTrasera()`). Mientras ésto sucede abre la puerta delantera (`abrirPuertaDelantera()`) y las personas que esperaban en la parada comienzan a ascender en orden siempre y cuando haya capacidad (notar que a esta altura ya se sabe cuántos pasajeros van a bajar).

El colectivo **recibe como parámetro la capacidad** (cantidad de pasajeros) del colectivo, que comienza vacío, y **el identificador** del colectivo (entre 0 y $M-1$).

Al llegar a una parada, el colectivo, se detiene con `detener()`. **Si hay pasajeros esperando para bajar**, éste abre su puerta trasera para indicar que ya pueden hacerlo (`abrirPuertaTrasera()`). Mientras ésto sucede abre la puerta delantera (`abrirPuertaDelantera()`) y las personas que esperaban en la parada comienzan a ascender **en orden** siempre y cuando **haya capacidad** (notar que a esta altura ya se sabe cuántos pasajeros van a bajar).

Las personas proceden a subir y el colectivo los atiende de a uno marcando en la máquina con `marcarTarifa()`, pero nunca lo hace antes de que el anterior haya terminado de `pagarConSUBE()`. Si no hay más pasajeros para subir o se llegó al límite de capacidad, el colectivo no duda en `cerrarPuertaDelantera()`, impidiendo que el resto de las personas en la parada ascienda. Una vez que los pasajeros terminan de ascender, el chofer espera a que terminen de descender todos los pasajeros que así lo desean, procede a `cerrarPuertaTrasera()` y `avanzar()` nuevamente a la siguiente parada, donde la dinámica será la misma.

Las personas proceden a subir y el colectivo los atiende **de a uno** marcando en la máquina con `marcarTarifa()`, pero **nunca lo hace antes de que** el anterior haya terminado de pagar `ConSUBE()`. Si no hay más pasajeros para subir o se llegó al límite de capacidad, el colectivo no duda en `cerrarPuertaDelantera()`, impidiendo que el resto de las personas en la parada ascienda. Una vez que los pasajeros terminan de ascender, el chofer **espera a que terminen de descender** todos los pasajeros que así lo desean, procede a `cerrarPuertaTrasera()` y `avanzar()` nuevamente a la siguiente parada, donde la dinámica será la misma.

Las personas proceden a subir y el colectivo los atiende **de a uno** marcando en la máquina con `marcarTarifa()`, pero **nunca lo hace antes de que** el anterior haya terminado de pagar con `SUBER()`. Si no hay más pasajeros para subir o se llegó al límite de capacidad, el colectivo no duda en `cerrarPuertaDelantera()`, impidiendo que el resto de las personas en la parada ascienda. Una vez que los pasajeros terminan de ascender, el chofer **espera a que terminen de descender** todos los pasajeros que así lo desean, procede a `cerrarPuertaTrasera()` y `avanzar()` nuevamente a la siguiente parada, donde la dinámica será la misma.

Puede asumir:

- El colectivo se detiene en todas las paradas y abre la puerta delantera, sin importar si hay pasajeros esperando para bajar o subir.
- El cálculo de la cantidad máxima de pasajeros que el colectivo deja subir puede realizarse considerando la cantidad de pasajeros que ya se sabe que descenderán (no es un problema que brevemente se vea superada la capacidad del colectivo mientras se espera que terminen de descender).

2) Extraer la información relevante relativa a la sincronización

- Dos tipos de procesos distintos

- Dos tipos de procesos distintos
- Los pasajeros suben en orden

- Dos tipos de procesos distintos
- Los pasajeros suben en orden
- Los pasajeros, al subir, son atendidos de a uno por vez

- Dos tipos de procesos distintos
- Los pasajeros suben en orden
- Los pasajeros, al subir, son atendidos de a uno por vez
- Los pasajeros bajan de a uno, pero sin orden

- Dos tipos de procesos distintos
- Los pasajeros suben en orden
- Los pasajeros, al subir, son atendidos de a uno por vez
- Los pasajeros bajan de a uno, pero sin orden
- La cantidad de colectivos/paradas es estática

- Dos tipos de procesos distintos
- Los pasajeros suben en orden
- Los pasajeros, al subir, son atendidos de a uno por vez
- Los pasajeros bajan de a uno, pero sin orden
- La cantidad de colectivos/paradas es estática
- La cantidad de pasajeros es dinámica

- Dos tipos de procesos distintos
- Los pasajeros suben en orden
- Los pasajeros, al subir, son atendidos de a uno por vez
- Los pasajeros bajan de a uno, pero sin orden
- La cantidad de colectivos/paradas es estática
- La cantidad de pasajeros es dinámica
- El chofer no arranca hasta que no bajaron/subieron todos

- Dos tipos de procesos distintos
- Los pasajeros suben en orden
- Los pasajeros, al subir, son atendidos de a uno por vez
- Los pasajeros bajan de a uno, pero sin orden
- La cantidad de colectivos/paradas es estática
- La cantidad de pasajeros es dinámica
- El chofer no arranca hasta que no bajaron/subieron todos
- En principio, el ascenso y el descenso pueden ocurrir *en paralelo*.

3) Si el ejercicio lo permite, pensar una **primera** solución *modularmente*

Por ejemplo, en **este caso**, podemos pensar una solución para un único colectivo y después extenderla a cualquier cantidad arbitraria (pero estática).

4) Extender/completar nuestra solución *parcial*

5) Rapasar párrafo por párrafo del enunciado para ver que nuestra solución respete lo pedido