

Práctica 4 – Pasaje de Mensajes

CONSIDERACIONES PARA RESOLVER LOS EJERCICIOS DE PASAJE DE MENSAJES ASINCRÓNICO (PMA):

- Los canales son compartidos por todos los procesos.
- Cada canal es una cola de mensajes; por lo tanto, el primer mensaje encolado es el primero en ser atendido.
- Por ser PMA, el **send** no bloquea al emisor.
- Se puede usar la sentencia **empty** para saber si hay algún mensaje en el canal, pero no se puede consultar por la cantidad de mensajes encolados.
- Se puede utilizar el **if/do** no determinístico donde cada opción es una condición booleana donde se puede preguntar por variables locales y/o por **empty** de canales.
if (cond 1) -> Acciones 1;
□ (cond 2) -> Acciones 2;
....
□ (cond N) -> Acciones N;
end if
De todas las opciones cuya condición sea Verdadera elige una en forma no determinística y ejecuta las acciones correspondientes. Si ninguna es verdadera, sale del if/do sin ejecutar acción alguna.
- Se debe evitar hacer **busy waiting** siempre que sea posible (sólo hacerlo si no hay otra opción).
- En todos los ejercicios el tiempo debe representarse con la función **delay**.

1. Suponga que N clientes llegan a la cola de un banco y que serán atendidos por sus empleados. Analice el problema y defina qué procesos, recursos y canales/comunicaciones serán necesarios/convenientes para resolverlo. Luego, resuelva considerando las siguientes situaciones:
 - a. Existe un único empleado, el cual atiende por orden de llegada.
 - b. Ídem a) pero considerando que hay 2 empleados para atender, ¿qué debe modificarse en la solución anterior?
 - c. Ídem b) pero considerando que, si no hay clientes para atender, los empleados realizan tareas administrativas durante 15 minutos. ¿Se puede resolver sin usar procesos adicionales? ¿Qué consecuencias implicaría?
2. Se desea modelar el funcionamiento de un banco en el cual existen 5 cajas para realizar pagos. Existen P clientes que desean hacer un pago. Para esto, cada uno selecciona la caja donde hay menos personas esperando; una vez seleccionada, espera a ser atendido. En cada caja, los clientes son atendidos por orden de llegada por los cajeros. Luego del pago, se les entrega un comprobante. **Nota:** maximizar la concurrencia.

3. Se debe modelar el funcionamiento de una casa de comida rápida, en la cual trabajan 2 cocineros y 3 vendedores, y que debe atender a C clientes. El modelado debe considerar que:

- Cada cliente realiza un pedido y luego espera a que se lo entreguen.
- Los pedidos que hacen los clientes son tomados por cualquiera de los vendedores y se lo pasan a los cocineros para que realicen el plato. Cuando no hay pedidos para atender, los vendedores aprovechan para reponer un pack de bebidas de la heladera (tardan entre 1 y 3 minutos para hacer esto).
- Repetidamente cada cocinero toma un pedido pendiente dejado por los vendedores, lo cocina y se lo entrega directamente al cliente correspondiente.

Nota: maximizar la concurrencia.

4. Simular la atención en un locutorio con 10 cabinas telefónicas, el cual tiene un empleado que se encarga de atender a N clientes. Al llegar, cada cliente espera hasta que el empleado le indique a qué cabina ir, la usa y luego se dirige al empleado para pagarle. El empleado atiende a los clientes en el orden en que hacen los pedidos. A cada cliente se le entrega un ticket factura por la operación.

- a) Implemente una solución para el problema descrito.
- b) Modifique la solución implementada para que el empleado dé prioridad a los que terminaron de usar la cabina sobre los que están esperando para usarla.

Nota: maximizar la concurrencia; suponga que hay una función *Cobrar()* llamada por el empleado que simula que el empleado le cobra al cliente.

5. Resolver la administración de 3 impresoras de una oficina. Las impresoras son usadas por N administrativos, los cuales están continuamente trabajando y cada tanto envían documentos a imprimir. Cada impresora, cuando está libre, toma un documento y lo imprime, de acuerdo con el orden de llegada.

- a) Implemente una solución para el problema descrito.
- b) Modifique la solución implementada para que considere la presencia de un director de oficina que también usa las impresas, el cual tiene prioridad sobre los administrativos.
- c) Modifique la solución (a) considerando que cada administrativo imprime 10 trabajos y que todos los procesos deben terminar su ejecución.
- d) Modifique la solución (b) considerando que tanto el director como cada administrativo imprimen 10 trabajos y que todos los procesos deben terminar su ejecución.
- e) Si la solución al ítem d) implica realizar *Busy Waiting*, modifíquela para evitarlo.

Nota: ni los administrativos ni el director deben esperar a que se imprima el documento.

CONSIDERACIONES PARA RESOLVER LOS EJERCICIOS DE PASAJE DE MENSAJES SINCRÓNICO (PMS):

- Los canales son punto a punto y no deben declararse.
- No se puede usar la sentencia *empty* para saber si hay algún mensaje en un canal.
- Tanto el envío como la recepción de mensajes es bloqueante.
- Sintaxis de las sentencias de envío y recepción:
Envío: nombreProcesoReceptor!port (datos a enviar)
Recepción: nombreProcesoEmisor?port (datos a recibir)

El port (o etiqueta) puede no ir. Se utiliza para diferenciar los tipos de mensajes que se podrían comunicarse entre dos procesos.

- En la sentencia de comunicación de recepción se puede usar el comodín * si el origen es un proceso dentro de un arreglo de procesos. Ejemplo: Clientes[*]?port(datos).
- Sintaxis de la Comunicación guardada:
Guarda: (condición booleana); sentencia de recepción → sentencia a realizar

Si no se especifica la condición booleana se considera verdadera (la condición booleana sólo puede hacer referencia a variables locales al proceso).

Cada guarda tiene tres posibles estados:

Elegible: la condición booleana es verdadera y la sentencia de comunicación se puede resolver inmediatamente.

No elegible: la condición booleana es falsa.

Bloqueada: la condición booleana es verdadera y la sentencia de comunicación no se puede resolver inmediatamente.

Sólo se puede usar dentro de un *if* o un *do* guardado:

El *if* funciona de la siguiente manera: de todas las guardas *elegibles* se selecciona una en forma no determinística, se realiza la sentencia de comunicación correspondiente, y luego las acciones asociadas a esa guarda. Si todas las guardas tienen el estado de *no elegibles*, se sale sin hacer nada. Si no hay ninguna guarda elegible, pero algunas están en estado *bloqueado*, se queda esperando en el *if* hasta que alguna se vuelva elegible.

El *do* funciona de la siguiente manera: sigue iterando de la misma manera que el *if* hasta que todas las guardas hasta que todas las guardas sean *no elegibles*.

1. Suponga que existe un antivirus distribuido que se compone de R procesos robots Examinadores y 1 proceso Analizador. Los procesos Examinadores están buscando continuamente posibles sitios web infectados; cada vez que encuentran uno avisan la dirección y luego continúan buscando. El proceso Analizador se encarga de hacer todas las

pruebas necesarias con cada uno de los sitios encontrados por los robots para determinar si están o no infectados.

- a) Analice el problema y defina qué procesos, recursos y comunicaciones serán necesarios/convenientes para resolverlo.
 - b) Implemente una solución con PMS sin tener en cuenta el orden de los pedidos.
 - c) Modifique el inciso (b) para que el Analizador resuelva los pedidos en el orden en que se hicieron.

2. En un laboratorio de genética veterinaria hay 3 empleados. El primero de ellos continuamente prepara las muestras de ADN; cada vez que termina, se la envía al segundo empleado y vuelve a su trabajo. El segundo empleado toma cada muestra de ADN preparada, arma el set de análisis que se deben realizar con ella y espera el resultado para archivarlo. Por último, el tercer empleado se encarga de realizar el análisis y devolverle el resultado al segundo empleado.

3. En un examen final hay N alumnos y P profesores. Cada alumno resuelve su examen, lo entrega y espera a que alguno de los profesores lo corrija y le indique la nota. Los profesores corrigen los exámenes respetando el orden en que los alumnos van entregando.
 - a) Considerando que $P=1$.
 - b) Considerando que $P>1$.
 - c) Ídem b) pero considerando que los alumnos no comienzan a realizar su examen hasta que todos hayan llegado al aula.

Nota: maximizar la concurrencia; no generar demora innecesaria; todos los procesos deben terminar su ejecución

4. En una exposición aeronáutica hay un simulador de vuelo (que debe ser usado con exclusión mutua) y un empleado encargado de administrar su uso. Hay P personas que esperan a que el empleado lo deje acceder al simulador, lo usa por un rato y se retira.
 - a) Implemente una solución donde el empleado sólo se ocupa de garantizar la exclusión mutua (sin importar el orden).
 - b) Modifique la solución anterior para que el empleado los deje acceder según el orden de su identificador (hasta que la persona i no lo haya usado, la persona $i+1$ debe esperar).
 - c) Modifique la solución a) para que el empleado considere el orden de llegada para dar acceso al simulador.

Nota: cada persona usa sólo una vez el simulador.

5. En un estadio de fútbol hay una máquina expendedora de gaseosas que debe ser usada por E Espectadores de acuerdo con el orden de llegada. Cuando el espectador accede a la máquina en su turno usa la máquina y luego se retira para dejar al siguiente. **Nota:** cada Espectador una sólo una vez la máquina.