

Práctica 4 – Pasaje de Mensajes

CONSIDERACIONES PARA RESOLVER LOS EJERCICIOS DE 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 pasaje de mensajes asincrónico 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 si hacer nada.
- Se debe tratar de evitar hacer **busy waiting** (sólo hacerlo si no hay otra opción).
- En todos los ejercicios el tiempo debe representarse con la función **delay**.

1. Se desea modelar el funcionamiento de un banco en el cual existen 5 cajas para realizar pagos. Existen P personas que desean pagar. Para esto cada una selecciona la caja donde hay menos personas esperando, una vez seleccionada espera a ser atendido.
2. Resolver la administración de las impresoras de una oficina. Hay 3 impresoras, N usuarios y 1 director. Los usuarios y el director están continuamente trabajando y cada tanto envían documentos a imprimir. Cada impresora, cuando está libre, toma un documento y lo imprime, de acuerdo al orden de llegada, pero siempre dando prioridad a los pedidos del director. **Nota:** los usuarios y el director no deben esperar a que se imprima el documento.
3. En una oficina hay **un empleado** para atender a **N personas**. Las personas pueden tener prioridad ALTA o BAJA (cada uno conoce su prioridad). El empleado atiende a las personas de acuerdo a la prioridad. Cada persona espera hasta que el empleado lo termina de atender y se retira. **Nota:** no debe hacerse *Busy Waiting*.
4. En un consultorio hay un médico que debe atender a 15 pacientes de acuerdo al turno de cada uno de ellos. Cada paciente ya conoce su turno al comenzar (valor entero entre 1 y 15), al llegar espera hasta que el médico lo llame para ser atendido y luego espera hasta que el médico lo termine de atender. **Nota:** los únicos procesos que se pueden usar son los que representen a los pacientes y al médico; se debe evitar hacer *Busy Waiting*.

5. Se debe modelar una casa de Comida Rápida, en el cual trabajan 2 cocineros y 3 vendedores. Además, hay C clientes que dejan un pedido y quedan esperando a que se lo alcancen. 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.
6. En una muestra de autos hay 5 simuladores para ser usados por N *personas* que asisten al evento. Hay un empleado que se encarga de administrar el uso de los mismos respetando el orden de llegada. Cuando una persona quiere usar un simulador debe esperar a que el empleado le indique uno LIBRE al cual ir, lo usa y luego le avisa al empleado que termino de usarlo. **Nota:** maximizar la concurrencia; suponga que hay una función $Usar(IdSim)$ llamada por las personas que simula el uso del simulador $IdSim$; todos los procesos deben terminar.
7. En una empresa hay *un empleado* para atender consultas realizadas por mail de C *clientes*. Los clientes hacen sucesivas consultas enviando por mail la consulta y esperando la respuesta. El empleado atiende las consultas respetando el orden en que se hicieron, y en aquellos momentos que no hay consultas para atender lee un libro por 20 minutos. **Nota:** todos los procesos están continuamente ejecutando.
8. Hay E *estudiantes* que rinden un examen final y *un profesor* que cuando todos los estudiantes han llegado les entrega el enunciado. Luego el profesor va corrigiendo los exámenes y enviando la nota de acuerdo con orden en que se van entregando. Cada alumno realiza el examen, lo entrega y espera a que el profesor le indique la nota. **Nota:** maximizar la concurrencia; no realizar *busy waiting*; todos los procesos deben terminar.
9. En un negocio de software que sólo atiende de forma virtual hay *un empleado* para atender pedidos de N *clientes*. Los pedidos pueden ser para hacer una COMPRA de un programa o para hacer una CONSULTA. El empleado atiende los pedidos priorizando los que son para COMPRAS: si es una COMPRA, prepara y configura el programa y le envía al cliente el código de acceso; en caso de ser una consulta le responde lo consultado. **Nota:** la persona sabe de qué tipo es cada pedido que realiza; ningún proceso debe terminar su ejecución; no realizar *busy waiting*.
10. En una empresa de software hay N *personas* que prueban un nuevo producto para encontrar errores, cuando encuentran uno generan un reporte para que uno de los **3 empleados** corrija el error y le avise cuando se haya terminado de resolver. Los empleados resuelven los reportes de acuerdo con el orden de llegada; cuando no hay reportes para atender los empleados se dedican a leer durante 10 minutos. **Nota:** todos los procesos están continuamente ejecutando; no generar demora innecesaria.

CONSIDERACIONES PARA RESOLVER LOS EJERCICIOS DE 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**.

11. En un laboratorio de genética veterinaria hay 3 empleados. El primero de ellos se encarga de preparar las muestras de ADN lo más rápido posible; el segundo, toma cada muestra de ADN preparada y arma el set de análisis que se deben realizar con ella y espera el resultado para archivarlo y continuar trabajando; el tercer empleado se encarga de realizar el análisis y devolverle el resultado al segundo empleado.

12. Suponga que existe un antivirus distribuido en él hay R procesos robots que continuamente están buscando posibles sitios web infectados; cada vez que encuentran uno avisan la dirección y continúan buscando. Hay un proceso analizador que se encargue de hacer todas las pruebas necesarias con cada uno de los sitios encontrados por los robots para determinar si están o no infectados.
13. En un banco hay un **empleado** para cobrar a N **personas**. Las N personas forman una única fila y el empleado los atiende de acuerdo al orden en que llegaron. Cuando una persona es llamada por el cajero, le entrega la boleta a pagar y este le devuelve la boleta sellada. **Notas:** cada persona paga una sola boleta.
14. En un examen final hay P alumnos y 3 profesores. Cuando todos los alumnos han llegado comienza el examen. 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. **Nota:** maximizar la concurrencia; no generar demora innecesaria; todos los procesos deben terminar.
15. En un estadio de fútbol hay una máquina expendedora de gaseosas que debe ser usada por **E Espectadores** (una vez vcada uno) de acuerdo al orden de llegada. Cuando el espectador accede a la máquina, la usa y luego se retira para dejar al siguiente.
16. En una empresa de software hay **3 programadores** que deben arreglar errores informados por **N clientes**. Los clientes continuamente están trabajando, y cuando encuentran un error envían un reporte a la empresa para que lo corrija (no tienen que esperar a que se resuelva). Los programadores resuelven los reclamos de acuerdo al orden de llegada, y si no hay reclamos pendientes trabajan durante una hora en otros programas. **Nota:** los procesos están continuamente ejecutando.
17. En una salita médica hay *un médico* para atender a *15 pacientes*. El médico atiende a los pacientes en el orden dado por el ID y les indica el tratamiento a realizar; asegurándose de que no haya dos pacientes al mismo tiempo en el consultorio. **Nota:** todos los procesos deben terminar; no usar procesos de más.
18. Se debe simular un partido de fútbol 11. Cuando los **22 jugadores** han llegado a la cancha se juega el partido durante 90 minutos y luego todos se retiran. **Nota:** todos los procesos deben terminar; evitar la demora innecesaria.
19. Hay E *estudiantes* que rinden un examen final y *un profesor* que les entrega el enunciado, y luego corrige los exámenes. Cada alumno espera a que le den el enunciado, y luego realiza su examen, al terminar lo entrega y espera a que el profesor le indique la nota. El profesor corrige los exámenes de a uno a la vez de acuerdo con el orden en que entregaron. **Nota:** todos los procesos deben terminar.
20. En una oficina hay **tres empleados** para atender a N **personas** que vienen a realizar un trámite. Cada persona espera a que cualquier empleado atienda su trámite y le dé el resultado. Cada empleado, cuando está libre atiende el siguiente pedido pendiente.