

Práctica 2 – Semáforos

CONSIDERACIONES PARA RESOLVER LOS EJERCICIOS:

- Los semáforos deben estar declarados en todos los ejercicios.
- Los semáforos deben estar inicializados en todos los ejercicios.
- No se puede utilizar ninguna sentencia para **setear** o **ver** el valor de un semáforo.
- Debe evitarse hacer **busy waiting** en todos los ejercicios.
- En todos los ejercicios el tiempo debe representarse con la función **delay**.

- 1) Un sistema operativo mantiene 5 instancias de un recurso almacenadas en una cola, cuando un proceso necesita usar una instancia del recurso la saca de la cola, la usa y cuando termina de usarla la vuelve a depositar.
- 2) Se tiene un curso con 40 alumnos, la maestra entrega una tarea distinta a cada alumno, luego cada alumno realiza su tarea y se la entrega a la maestra para que la corrija, esta revisa la tarea y si está bien le avisa al alumno que puede irse, si la tarea está mal le indica los errores, el alumno corregirá esos errores y volverá a entregarle la tarea a la maestra para que realice la corrección nuevamente, esto se repite hasta que la tarea no tenga errores.
- 3) Existen N personas que deben fotocopiar un documento cada una. Resolver cada ítem usando semáforos:
 - a) Implemente una solución suponiendo que existe una única fotocopidora compartida por todas las personas, y las mismas la deben usar de a una persona a la vez, sin importar el orden. Existe una función *Fotocopiar()* llamada por la persona que simula el uso de la fotocopidora. Sólo se deben usar los procesos que representan a las *Personas*.
 - b) Modifique la solución de (a) para el caso en que se deba respetar el orden de llegada.
 - c) Modifique la solución de (b) para el caso en que se deba dar prioridad de acuerdo a la edad de cada persona (cuando la fotocopidora está libre la debe usar la persona de mayor edad entre las que estén esperando para usarla).
 - d) Modifique la solución de (a) para el caso en que se deba respetar estrictamente el orden dado por el identificador del proceso (la persona X no puede usar la fotocopidora hasta que no haya terminado de usarla la persona X-1).
- 4) Una empresa de turismo posee 4 *combis* con capacidad para 25 personas cada una. Hay un único *empleado* para vender pasajes en cualquiera de las combis, el cual debe atender de acuerdo con el orden de llegada a N *clientes*. Cada cliente va a comprar un pasaje en una combi en particular (el cliente conoce para que combi quiere comprar pasaje), espera que le den el pasaje, sube a la combi y luego espera a llegar a destino para bajarse. Cada combi espera a que suban los 25 pasajeros que le corresponden para realizar el viaje. **Nota:** maximizar la concurrencia

- a) Suponga que $N=100$ y hay exactamente 25 clientes para cada combi.
- b) Suponga que $N > 100$ y hay AL MENOS 25 clientes para cada combi. Si el cliente quiere comprar pasaje para una combi que ya vendió sus 25 pasajes se retira sin viajar.
- 5) Se debe simular una maratón con **C corredores** donde en la llegada hay UNA máquina expendedoras de agua con capacidad para 20 botellas. Además, existe **un repositor** encargado de reponer las botellas de la máquina. Cuando los **C corredores** han llegado al inicio comienza la carrera. Cuando un corredor termina la carrera se dirigen a la máquina expendedora, espera su turno (respetando el orden de llegada), saca una botella y se retira. Si encuentra la máquina sin botellas, le avisa al **repositor** para que cargue nuevamente la máquina con 20 botellas; espera a que se haga la recarga; saca una botella y se retira. **Nota:** maximizar la concurrencia; mientras se reponen las botellas se debe permitir que otros corredores se encolen.
- 6) A una cerealera van T camiones a descargarse trigo y M camiones a descargar maíz. Sólo hay lugar para que 7 camiones a la vez descarguen, pero no pueden ser más de 5 del mismo tipo de cereal. **Nota:** sólo se pueden usar procesos que representen a los camiones.
- 7) 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:** todos los procesos deben terminar.
- 8) En un supermercado hay 2 cajas para cobrar a los C clientes, una para quienes llevan menos de 20 productos y otra para el resto. En cada caja hay un empleado para cobrar a los clientes correspondientes de acuerdo con el orden de llegada. **Nota:** maximizar la concurrencia; suponga que existe la función *Atender()* que simula la atención por parte del empleado; el empleado no puede atender a un cliente hasta que no se haya ido el anterior.
- 9) Simular la atención en una Terminal de Micros que posee 3 puestos para hisopar a **150 pasajeros**. En cada puesto hay **una Enfermera** que atiende a los pasajeros de acuerdo al orden de llegada al mismo. Cuando llega un pasajero se dirige al puesto que tenga menos gente esperando. Espera a que la enfermera correspondiente lo llame para hisoparlo, y luego se retira. **Nota:** suponer que existe una función *Hisopar()* que simula la atención del pasajero por parte de la enfermera correspondiente.
- 10) 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 no deben terminar (trabajan en un loop infinito); suponga que hay una función *ResolverError* que simula que un programador está resolviendo un reporte de un cliente, y otra *Programar* que simula que está trabajando en otro programa.
- 11) En un supermercado hay una caja autónoma (sin un empleado que la atienda) que es usada

por los clientes para pagar las compras realizadas. Hay N *clientes* que la deben usar de acuerdo con el orden de llegada (suponga que existe una función *UsarCaja()* que simula este uso).

Nota: sólo se pueden usar los procesos que representan a los clientes.

- 12) En una competencia de programadores hay **10 participantes** y **UN coordinador**. El coordinador evalúa a cada participante de a uno a la vez de acuerdo con el orden de llegada. Para evaluar a un participante el coordinador le entrega el problema a resolver y espera la resolución para corregirla y darle el puntaje al participante. **Nota:** el coordinador no debe comenzar a evaluar a un participante hasta que el anterior no se haya ido; todos los procesos deben terminar.
- 13) Se tiene un vector A de 1.000.000 de números, del cual se debe obtener la suma de todos los valores utilizando **5 procesos Worker**. Al terminar todos los procesos deben imprimir el resultado final. **Nota:** maximizar concurrencia; únicamente se pueden usar los 5 procesos *Worker*.
- 14) Existen **B barcos** areneros que deben descargar su contenido en una playa. Los barcos se descargan de a uno por vez, y de acuerdo con orden de llegada. Una vez que el barco llegó, espera a que le llegue su turno para comenzar a descargar su contenido, y luego se retira. **Nota:** sólo se pueden usar los procesos que representen a los barcos; cada barco descarga sólo una vez; suponga que existe la función *DESCARGAR()* que simula que el barco está descargando su contenido en la playa.