

**Ejercicio 1:** Resolver el problema de los Productores / Consumidores mediante la utilización de paso de mensajes SIN tamaño de “buffer”, es decir “buffer” ilimitado, el objeto enviado al “buffer” es un String y no importa su contenido.

*Nota:* Utilizar utilizar send no bloqueante y receive bloqueante.

**Ejercicio 2:** Resolver el problema de los Productores / Consumidores mediante la utilización de paso de mensajes SIN tamaño de “buffer”, es decir “buffer” ilimitado, el objeto enviado al “buffer” es un Item, definir el mismo con dos atributos: descripción y cantidad.

*Nota:* Utilizar utilizar send no bloqueante y receive bloqueante.

**Ejercicio 3:** Resolver el ejercicio del contador (un hilo suma 5000 y otro suma 3000 y a veces el resultado es menor a 8000) realizando una mutua exclusión mediante paso de mensajes.

**Ejercicio 4:** Resolver el ejercicio del parcial de los tenores mediante paso de mensajes.

*Nota:* Solo se puede utilizar un “buzon”.

**Ejercicio 5:** Resolver el problema de los Productores / Consumidores mediante la utilización de paso de mensajes CON tamaño de “buffer”, es decir “buffer” Limitado a cinco posiciones, el objeto enviado al “buffer” es un Item, definir el mismo con dos atributos: descripción y cantidad.

**Ejercicio 6:** Resolver el problema de los Lectores / Escritores con prioridad Lectura mediante paso de mensajes.

**Ejercicio 7:** En el ejercicio 1 convertir el “receive” del Consumidor en “no bloqueante”, pero mediante la utilización de monitores hacer que este hilo funcionara como si fuera bloqueante, es decir, cuando el mismo comienza a ejecutar quede suspendido, luego el mismo es liberado cuando el Listener recibe el mensaje.

**Ejercicio 8:** Mediante el estilo de mensajería “publish & subscribe” realizar un ejemplo que contenga dos “Topic” y que cada “Topic” tenga sus subscriptores (más de uno), luego cuando el publicador de cada “Topic” publique un mensaje ver que sus respectivos subscriptores lo reciban.