



TP6: Ejercicios varios multiprocessor y multicomputer.

Bibliografía: [Lester93] Bruce P. Lester. *The Art of Parallel Programming*. Prentice Hall, 1993.

1. Una matriz de $n \times n$, M , se dice **equi-transpuesta** si M y la transpuesta de M son matrices idénticas.

Para una arquitectura **con topología RING** escriba un programa en Multi-Pascal que, utilizando un algoritmo con paralelismo, indique si una matriz M es equi-transpuesta. En base a su solución, indique en función de la dimensión de la matriz (n) cuál es el número mínimo de procesadores que necesita.

1	3	7
3	5	2
7	2	8

Es equi-transpuesta

1	3	7
3	5	2
1	3	9

No es equi-transpuesta

2. Dada una matriz de $n \times n$, M , y un vector de n elementos, A , se debe construir una matriz resultado R tal que:

$$R[i, j] = R'[i, j] + R'[j, i]$$

$$R'[i, j] = A[i] * \sum_{k=i}^n M[k, j] + A[j]$$

Por ejemplo: $R'[1,1] = 3*(4+6+5+1) + 3 = 51$; $R[1,1] = R'[1,1] + R'[1,1] = 102$;
 $R'[2,3] = 6*(2+6+4) + 7 = 79$; $R[2,3] = R'[2,3] + R'[3,2] = 79 + 69 = 148$;
 $R'[3,2] = 7*(9+0) + 6 = 69$; $R[3,2] = R'[3,2] + R'[2,3] = 69 + 79 = 148$;

3
6
7
12

op

4	2	0	8
6	3	2	1
5	9	6	5
1	0	4	7

=

102			
		151	
	151		

A
M
R

[En su programa no necesita implementar el ingreso de datos ni la impresión en pantalla]

Para una arquitectura **con topología MESH** escriba un programa en Multi-Pascal que, utilizando un algoritmo con paralelismo, calcule la matriz R a partir de M y A . En base a su solución, indique en función de la dimensión de la matriz (n) cuál es el número mínimo de procesadores que necesita.



3. Considere las siguientes declaraciones donde se define una lista enlazada con punteros donde cada celda de la lista contiene un arreglo de enteros:

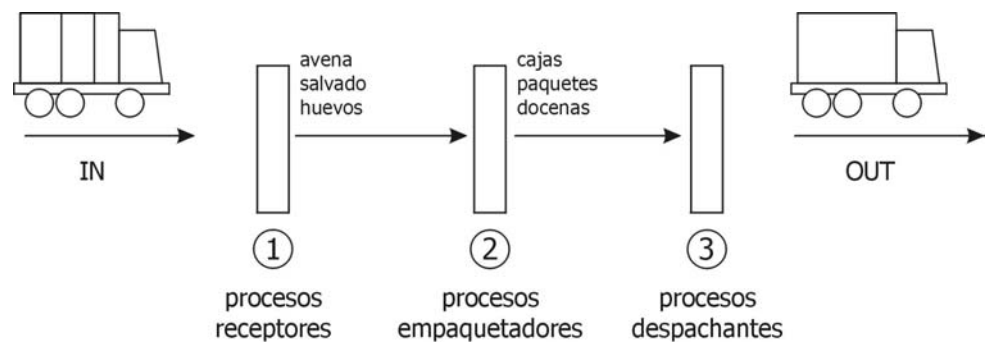
```
CONST    MAX = ...;  
         RUEDA = 1;  
         CARROCERIA = 2;  
  
TYPE TARREGLO = ARRAY [1..MAX] OF INTEGER;  
         LISTA = ^CELDA;  
         CELDA = RECORD  
             ELEM: TARREGLO;  
             SIG: LISTA;  
         END;
```

Se tiene una línea de ensamble representada por esta lista enlazada de elementos. La lista contiene lotes (grupos) de elementos, dados por los arreglos. Cada elemento del arreglo consiste de una pieza de auto: una **rueda** o una **carrocería**. Cada auto está formado por una carrocería y cuatro ruedas. La cantidad de ruedas es suficiente para armar los autos.

Además, existen dos clases de robots. Los robots **clasificadores** procesan el lote que les toca y van separando los elementos según su tipo. Hay 5 (cinco) robots **armadores** fijos que ensamblan las cuatro ruedas en la carrocería y arman el auto. Cada armador lleva la cuenta de la cantidad de autos que armó.

Para una arquitectura de **memoria compartida** escriba un programa en Multi-Pascal que, utilizando un algoritmo con paralelismo, simule la tarea descrita, procesando la lista de elementos, clasificando y ensamblando los autos. La condición de terminación es que todos los robots clasificadores y armadores hayan terminado. Finalmente, el programa debe imprimir la cantidad total de autos ensamblados.

4. Existe una fábrica empaquetadora a la que **ingresan camiones con avena, salvado y/o huevos** a granel. También ingresan camiones con pedidos para realizar repartos.





Paralelismo y Concurrencia en Sistemas

Prof. Alejandro J. García
Asist. Natalia L. Weinbach



Los camiones que entran son recibidos por 3 (tres) **procesos receptores** que leen de un canal el contenido de los camiones y dirigen su carga a los **depósitos adecuados**. Luego, existen **procesos empaquetadores** que de de acuerdo a los pedidos de los camiones de reparto arman cajas de avena con 500gr. de contenido; paquetes de salvado también de 500gr. y cartones con docenas de huevos. Además, se producen paquetes de cereal de 400gr., con una mezcla en partes iguales de avena y salvado.

Esta mercadería empaquetada es manejada por 5 (cinco) **procesos despachantes** que llenan a los **camiones de reparto** de acuerdo a sus **pedidos de carga**.

Para una arquitectura de **memoria compartida** escriba un programa en Multi-Pascal que, utilizando un algoritmo con paralelismo, implemente los procesos receptores, empaquetadores y despachantes para simular la tarea descrita, procesando en un ciclo continuo los camiones que entran y cargando la mercadería empaquetada en los camiones salientes según lo indicado. Los camiones de reparto salientes deben retenerse la menor cantidad de tiempo posible, dado que la rapidez de la entrega es muy importante.

5. El Call Center de una reconocida casa de repuestos de Córdoba atiende requerimientos de todo el país. Cada día, de 8 a 20hs, se atiende una serie de llamados telefónicos y se los procesa de acuerdo al producto. Al final del día salen los camiones con distinto destino a realizar los repartos.

Existen **cuatro centrales de atención telefónica** para las distintas zonas. Si el llamado arriba a la central 1, el **destino** es zona norte. A la central 2 le corresponde zona sur, y a las centrales 3 y 4 la zona oeste y este respectivamente.

Las **áreas** que se manejan son: electricidad, mecánica y accesorios; y cada área ofrece para la venta dos productos como se muestra a continuación. Los telefonistas atienden una **cantidad finita de llamados** que requieren una **cierta cantidad de determinado producto** y arman un pedido con una descripción de área y un destino.

Electricidad	Mecánica	Accesorios
control de calefacción levanta-vidrio eléctrico	radiador rulemanes	cubierta faro delantero

Luego, los pedidos ingresan al **salón comercial** donde cada área tiene una serie de personas responsables que toman el pedido y lo arman. Cada área cuenta con un pequeño **stock** de cada producto. Si en cualquier momento se produce una falta en el stock, se puede solicitar una reposición al **depósito** donde siempre hay mercadería. Las reposiciones de productos se realizan en lotes de 10.

Cuando el pedido está listo, se carga en alguno de los camiones disponibles de acuerdo a su destino. En **despacho** hay cuatro colas de camiones, una para cada zona, capaces de transportar hasta 20 pedidos cada uno. Se desea llevar una contabilidad de la cantidad de camiones que salieron hacia cada destino para realizar un estudio de mercado.



Paralelismo y Concurrencia en Sistemas

Prof. Alejandro J. García
Asist. Natalia L. Weinbach



Para una arquitectura de **memoria compartida** escriba un programa en Multi-Pascal que, utilizando un algoritmo con paralelismo, simule la tarea descrita, procesando los llamados, armando pedidos y despachando en los camiones. La condición de terminación se da cuando no hay más llamados pendientes por atender y no hay ningún pedido por procesar. Finalmente, el programa debe imprimir la cantidad total de camiones al norte, sur, este y oeste que son despachados durante un día de trabajo.

