Práctica 1

1) P1)

```
1- LOAD PosMemoriaY, reg1
2- ADD reg1, 2
3 -STORE PosMemoriaY, reg1
```

P2)

```
4- LOAD PosMemoriaY, reg2
5- ADD reg2, 1
6- STORE PosMemoriaX, reg2
```

P3)

```
7- LOAD PosMemoriaX, reg3
8- LOAD PosMemoria X, regAcumulador
9- MUL reg3, 3
10- LOAD regAcumulador, reg3
11- MUL regAcumulador, 2
12- ADD reg3, regAcumulador
13- ADD reg3, 1
14- STORE posMemoriaX, reg3
```

1) Da 56 si la historia es: 1,2,3,4,5,6,7,8,9,10,11,12,13,14

Me dio paja seguir con el punto 1

2)

```
//se puede hacer de esta forma
totalN=0
process contarN[id: 0.. J]{

        parcial:= parcial +1
```

```
    <totalN:= totalN+ parcial>
}

//se puede hacer de esta otra forma
Vector v[M];
totalN=0
process contarN[id:0.. J]{
  parcial=0
  for i:= id* (M/J) to (id *M/J + M/J) - 1
    if v[i]= N
      parcial=parcial + 1
  <totalN:= totalN + parcial
 }
```

3) Es erróneo ya que en la línea 3 del proceso Productor al incrementar cant, se va a cumplir la condición en el Consumidor y puede suceder que el consumidor tome un elemento que en realidad el productor todavía no colocó en el buffer ya que no se sabe si el productor va a hacerlo primero, lo mismo pasaría si un Consumidor produce otro elemento y lo deja antes de que el productor lo tome.

debería corregirse así:

```
int cant = 0; int pri_ocupada = 0; int pri_vacia = 0; int buf
Process Productor::
{ while (true)
 { produce elemento
 <await (cant < N); buffer[pri_vacia] = elemento; cant++>
 pri_vacia = (pri_vacia + 1) mod N;
 }
}
Process Consumidor::
{ while (true)
 { <await (cant > 0);  elemento = buffer[pri_ocupada]; cant--
 pri_ocupada = (pri_ocupada + 1) mod N;
 consume elemento
```

```
 }
}
```

b)

```
int cant = 0; int pri_ocupada = 0; int pri_vacia = 0; int buf
Process Productor[id:0..P]::
{ while (true)
 { produce elemento
 <await (cant < N); buffer[pri_vacia] = elemento;  pri_vacia
 }
}
Process Consumidor[id:0..C]::
{ while (true)
 { <await (cant > 0);  elemento = buffer[pri_ocupada]; pri_oc
 consume elemento
 }
}
```

4.

```
int N=5;
cola [N];
cant:= 5
//siempre se pone [id: 0..N] aunque no me diga cuantos son?

proces sacarRecurso[id: 0.. N]{
    while(true)
        <await< (cant>=0) >; elemento=pop(cola, elemento); ca
        //usa el elemento
        <push(cola, elemento); cant++>
        }
```

5. A.

```
boolean usando= False
process persona[id: 0..N-1]{
```

```
    while(true)
     <persona[id].imprimir(documento)>
```

5.B

```
Cola cola;
siguiente=-1
process persona[id: 0.. N-1]{
    <if (siguiente = -1) siguiente=id; else push(cola, id);>
    <await(siguiente=id)>
    persona[id].imprimir
    <if (cola.vacía()) siguiente=-1; else pop(cola, siguiente
}
```

5.C.

```
Cola cola;
siguiente=-1
process persona[id: 0.. N-1]{
    <if (siguiente = -1) siguiente=id; else insertarOrdenado(
    <await(siguiente=id)>
    persona[id].imprimir
    <if (cola.vacía()) siguiente=-1; else siguiente=pop(cola,
}
```

5.D

```
boolean usando= False
Cola cola;
siguiente=-1
```

```
process persona[id: 0.. N-1]{
 while (true){
   <push(cola,id)>
    <await(siguiente=id); usando=True>
    persona[id].imprimir
    <usando:= False>
}

process coordinador(){
    while(true)
        <await(usando=False); if(cola.vacia()) siguiente= -1;
```

7)

```
int solicita [n]
process sc [i= 1 to n]{
  while(true){
    solicita [id]=1
    while (solicita[id]==1) skip;
    SC
    solicita[id]= -1
    SNC
  }
}

process coordinador{
  while(true){
    for [i= 1 to n]{
      if(solicita[i]==1){
        solicita[i]=0
        while (solicita [id]==0) skip;
        }
      }
    }
  }
```