

Acción atómica (básicas)

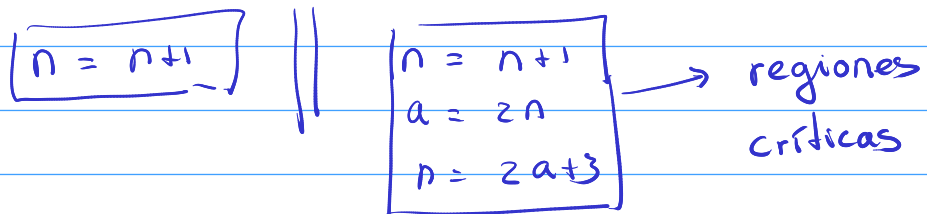
$$n = n + 1$$

no es atómico

↓

LOAD N
INC
STORE N

$$n = 0$$



P → a; b

Q → r; s; t

Modelo de interleaving (Entremezclado)

a; r; s; b; t

$$n = 0$$

$$n = n + 1$$

||

$$n = n + 1$$

$$n = 0$$

a LOAD N

b INC

c STORE N

||

LOAD N r

INC s

STORE N t

$n=1$ $n=2$
 $a; b; c; r; s; t$
 $r; s; t; a; b; c$

buena
buena

no hay
interrupciones

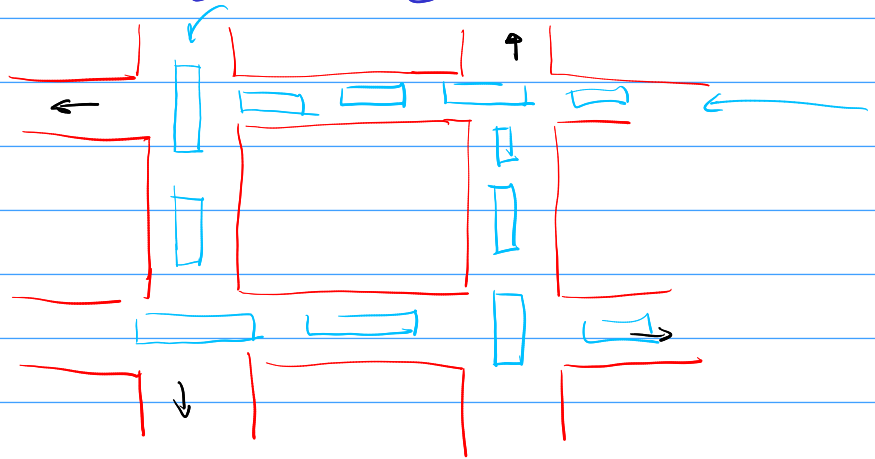
		N	REG
a	LOAD N	0	0
b	INE	0	1
r	LOAD N	0	0
s	INC	0	1
c	STORE N	1	1
t	STORE N	1	1

$N=1$

demostrar propiedades de programas

- Seguridad (safety)

- Exclusión mutua
- Ausencia de deadlocks

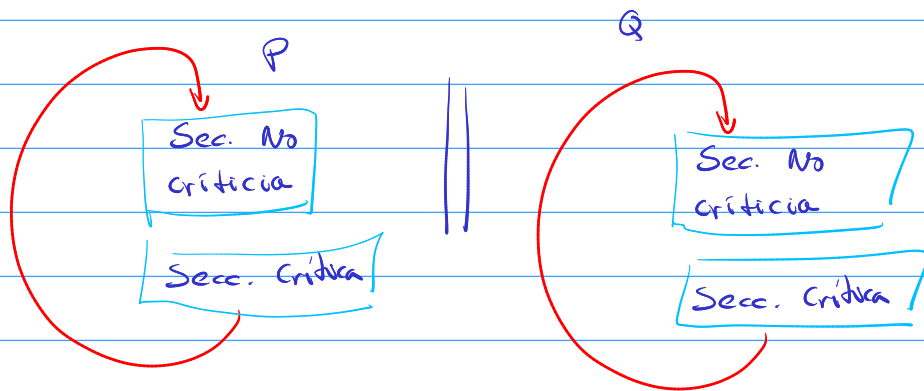


- Vivacidad (Liveness)

Todo acaba sucediendo

- Inanición
- Justicia (fairness)

Problema de la exclusión mutua



loop
 [Secc. No crítica] → se puede parar
 Pre-protocolo
 [Secc. Crítica] → No se puede parar
 Post-protocolo
end loop

- Exclusión mutua

P y Q no pueden ejecutar a la vez la sección crítica

- Ausencia de dead lock

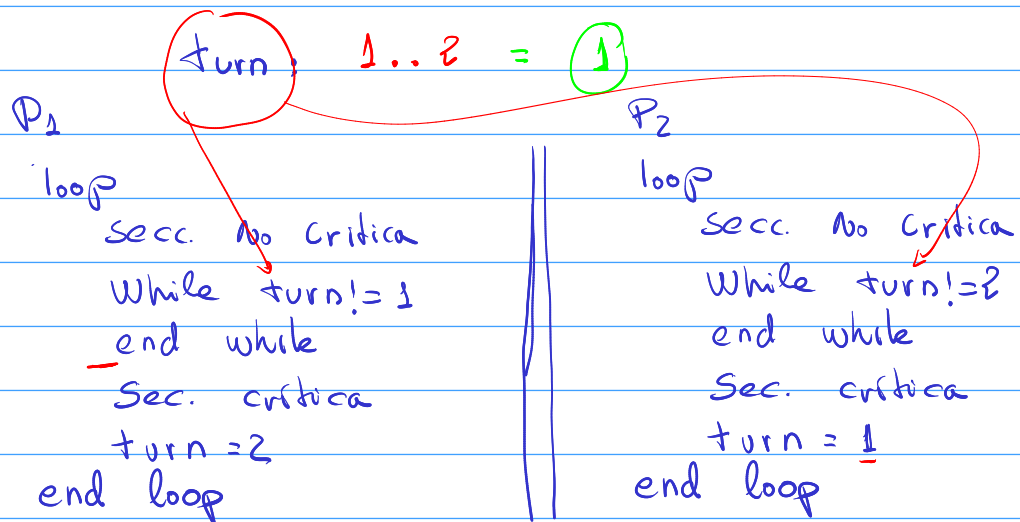
- No inanición

- Dinamismo en ausencia de competencia

Propuesta 1

P₁

P₂



Pros

- Exclusión mutua
- No hay bloqueos
- No hay inanición

Contra:

- Orden estricto
- No hay viveza sin competencia

$c_1, c_2 : Bool = False$

$c_1 \leftrightarrow P_1$ sección crítica
 $c_2 \leftrightarrow P_2$ sección crítica

P_1

```
loop
  secc. No crítica
a → while c2
    end while
b → c1 = True
    secc. crítica
    c1 = False
end loop
```

P_2

```
loop
  secc. No crítica
c | while c1
d | end while
  | c2 = True
  | secc. crítica
  | c2 = False
end loop
```

Favor

- Hay dinamismo
sin competencia

¿Exclusión mutua?

$c_1 \rightsquigarrow False$
 $c_2 \rightsquigarrow False$

$a; c; b; d$ ¿es posible?

c_1 y c_2 van True de forma
simultánea

No se garantiza exclusión mutua

Propuesta 3

$C1$ y $C2$ no sean True
de forma simultánea

$C1, C2: \text{Bool} = \text{False}$

P_1

loop

Secc. No Crítica

$C1 = \text{True}$

while $C2$

$C1 = \text{False}$
 $C1 = \text{True}$

end while

Secc. Crítica

$C1 = \text{False}$

end loop

$C1 = \text{True}$
 $C2 = \text{False}$

P_2

loop

Secc. No Crítica

$C2 = \text{True}$

while $C1$

$C2 = \text{False}$
 $C2 = \text{True}$

end while

Secc. Crítica

$C2 = \text{False}$

end loop

- Inanición global bnde intermedias
- Inanición de 1 proceso
(Secc. No crítica rápida)

Algoritmo de Decker

$C1, C2 : \text{Bool} = \text{False}$

$\text{Turn} : 1..2 = 1$

P_i en sección crítica $\rightarrow C_i$

P_1

```
loop
  Secc No Crítica
   $C1 = \text{True}$ 
  while  $C2$ :
     $C1 = \text{False}$ 
    while  $\text{turn} \neq 1$ :
    end while
     $C1 = \text{true}$ 
  end while
  Secc. crítica
   $C1 = \text{False}$ 
   $\text{turn} = 2$ 
end loop
```

P_2

```
loop
  Secc No Crítica
   $C2 = \text{True}$ 
  while  $C1$ :
     $C2 = \text{False}$ 
    while  $\text{turn} \neq 1$ :
    end while
     $C2 = \text{true}$ 
  end while
  Secc. crítica
   $C2 = \text{False}$ 
   $\text{turn} = 1$ 
end loop
```