

## Algoritmo incorrecto de Hyman

Ignacio Pérez Laborda  
Bárbara Martínez

Facultad de Tecnología Informática  
Universidad de Belgrano

# Introducción

- 1 Este algoritmo se pensó para darle una solución al problema de la exclusión mutua en la programación concurrente.
- 2 Harris Hyman, estudiante universitario, propuso esta solución en 1966 basándose en el existente algoritmo de Petersen.
- 3 Fue publicado y aprobado por varios revisores pero luego los publicantes se retractaron debido a que se descubrió que el algoritmo era incorrecto.

# Descripción

- \* Consiste en turnar la ejecución de dos procesos concurrentes, cuando uno esta trabajando, ciclará hasta que termine y luego pasará al proceso siguiente.
- \* Sin embargo el algoritmo presenta fallas en la resolución de la exclusión mutua.

## Código fuente del Algoritmo

```
void Protocol(int me, int you)
{
    do{
1      flag[me] = true;
2      while (turn != me){
3          while(flag[you])
4              /* do nothing */
5              turn = me;
        }
6      CriticalSection(me);
7      flag[me] = false;
8      RemainderSection(me);
    } while(true);
}
```

# Ejemplo

Supongamos que:

- \* La variable turn esta en 0
- \* Los flags[0] y flags[1] estan en falso
- \* Los procesos a ejecutar son 1,2,3,5 y 6:

El proceso P1 ejecuta 1, 2, 3. Como flag[0] es falso, la próxima instrucción que debe ejecutar P1 seria 5, pero lo hará próximamente.

P0 ejecuta 1, 2 y 6.

Por ultimo P0 quiere ejecutar 5 y 6 pero no puede porque P0 está trabajando en el proceso 6, ambos se encuentran en violación exclusión mutua.

# Ejemplo

Esta es una imagen demostrativa de la ejecución del algoritmo

P1		P0
1		flag[1]==true
2		
3		
		1
		flag[0]==true
		2
		6
5		turn==1
6		mutual exclusion has been violated

=====