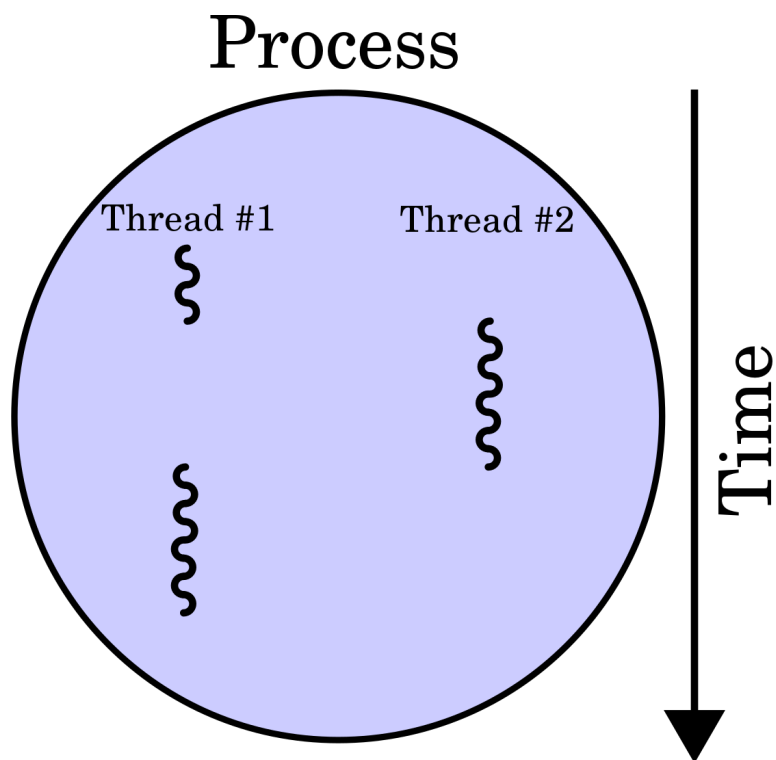


# SISTEMES OPERATIUS 2:

## Programació multifil: funcions de bloqueig



Lin Zhipeng  
Oscar de Caralt

Curs 2022-2023

## **Introducció:**

Aquesta pràctica està centrada en la utilització de les funcions de creació i bloqueig de fils utilitzant monitors per a la implementació de la solució.

## **Preguntes a respondre:**

- **1. Quines variables haurà de passar (per argument) el fil principal als fils secundaris?**

El File per a poder llegir les dades del fitxer, la matriu d'aeroports per a tenir la informació dels aeroports i la variable num\_flights per anar fent l'estadística del fitxer.

- **2. Els fils secundaris accedeixen a recursos (variables) compartits entre ells. Quines seran les seccions crítiques? Quines parts del codi són les que s'han de protegir? Cal protegir la lectura del fitxer? Cal protegir l'extracció de dades del bloc? Cal protegir l'actualització de la variable num\_flights? Comenteu la vostra resposta.**

Les seccions crítiques seran aquells blocs de codi on s'accedeix a recursos compartits i on, per tant, no hauria d'haver-hi més d'un fil d'execució simultàniament. Les parts del codi a protegir són aquelles on realitzem accions per intentar accedir/modificar els recursos compartits... Les lectures al nombre N de línies seguides d'un bloc de codi s'han de protegir per així evitar que cap altre fil llegeixi línies que ja han estat llegides per un altre fil. Tanmateix, les extraccions de dades del bloc han d'estar protegides. També s'ha de protegir l'actualització de la variable num\_flights per tal d'assegurar que el valor d'aquesta és correcte (que les dades i valors s'han llegit correctament).

### Exercicis a fer:

Taula que mostra el temps d'execució segons el nombre de fils (F) i el nombre de línies seguides d'un bloc que el fil llegeix (N) respecte el fitxer 2007.csv

<b>Temps en segons Fils(F) / Línies(N)</b>	<b>N=1</b>	<b>N=10</b>	<b>N=100</b>	<b>N=10.000</b>	<b>N=100.000</b>
<b>F = 1</b>	<b>14,67</b>	<b>14,49</b>	<b>14,29</b>	<b>14,33</b>	<b>14,43</b>
<b>F = 2</b>	<b>19,75</b>	<b>10,68</b>	<b>9,18</b>	<b>9,27</b>	<b>12,07</b>

Després de realitzar diversos experiments, hem arribat a la conclusió que és més eficient fer servir diversos fils (sempre que el nombre de línies seguides d'un bloc que el fil llegeixi sigui superior a 1, ja que sinó trigarà més perquè estem cridant massa cops a lock() i a unlock()) ja que permet aprofitar el temps d'inactivitat i com a resultat aconseguim reduir el temps d'execució del programa. Tot i això, no hem de sobrecarregar els fils amb una gran quantitat de línies a llegir perquè el resultat seria contraproductiu (comença a ser massa treball per a un fil i, per tant, és més lent). Per la qual cosa creiem que l'òptim és utilitzar diversos fils amb un nombre moderat de línies (ni moltes ni poques).

### Conclusions:

Després d'aquesta pràctica, queda demostrat com la programació paral·lela ens permet resoldre un problema més ràpid i eficientment, sempre que no ens sobrepassem o ens quedem curts amb els paràmetres que passem. A més a més, ens hem adonat de la importància de l'exclusió mútua i que cada fil necessita la seva pròpia clau per a la seva pròpia secció crítica.