



Universitat de Lleida  
Escola Politècnica Superior

07/02/2021

# Reentrega Pràctica 1

Grau en Enginyeria Informàtica

Joel Farré Cortés (78103400T)

Joel Aumedes Serrano (48051307Y)

ESCOLA POLITÈCNICA SUPERIOR – UNIVERSITAT DE LLEIDA

## Anàlisi del disseny de la nova versió

Per a poder obtenir una versió més eficient i ràpida busquem al codi seqüencial els llocs on es realitzen més càlculs i on es pot obtenir una major optimització. Aquest lloc és el càlcul de la validesa i el valor d'un equip. Al haver-hi un nombre tan gran d'equips possibles, un sol *thread* s'estava molta estona dins del bucle que els comprova.

La funció que realitza el programa en la nova versió concurrent és la de separar els equips de forma equitativa entre el número de *threads* que ha passat l'usuari i llavors es creen els *threads* que calculen quin és el millor equip dels que se'ls hi ha assignat.

Els *threads* al acabar de realitzar els seus càlculs, retornen un millor equip parcial. El *thread* principal és l'encarregat d'un cop haver acabat de calcular el seu millor parcial, comprar els millors parcials per a finalment determinar quin és el millor total.

## Anàlisi del rendiment

Per a dur a terme l'anàlisi del rendiment per a cada fitxer, s'ha agafat com a valors del pressupost 200 i com a nombre de *threads* 6. Un cop seleccionats aquest valors, es mostren els temps d'execució a la taula següent. Per a que les condicions siguin les mateixes, s'ha agafat com a referència de temps el valor obtingut en la primera execució.

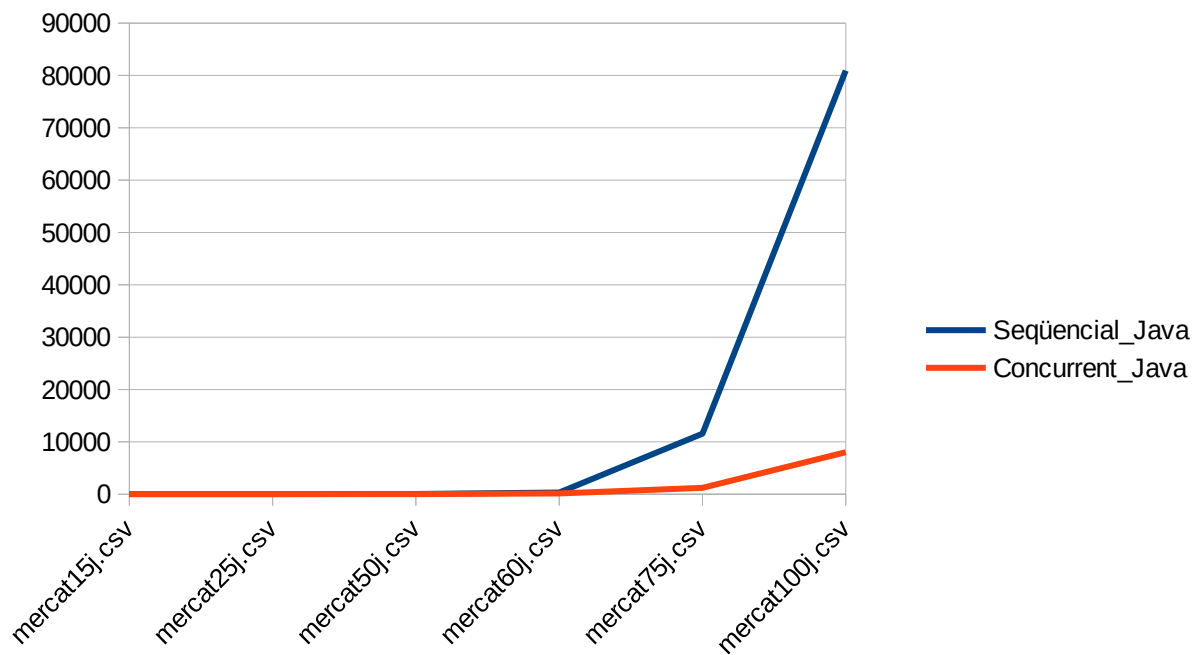
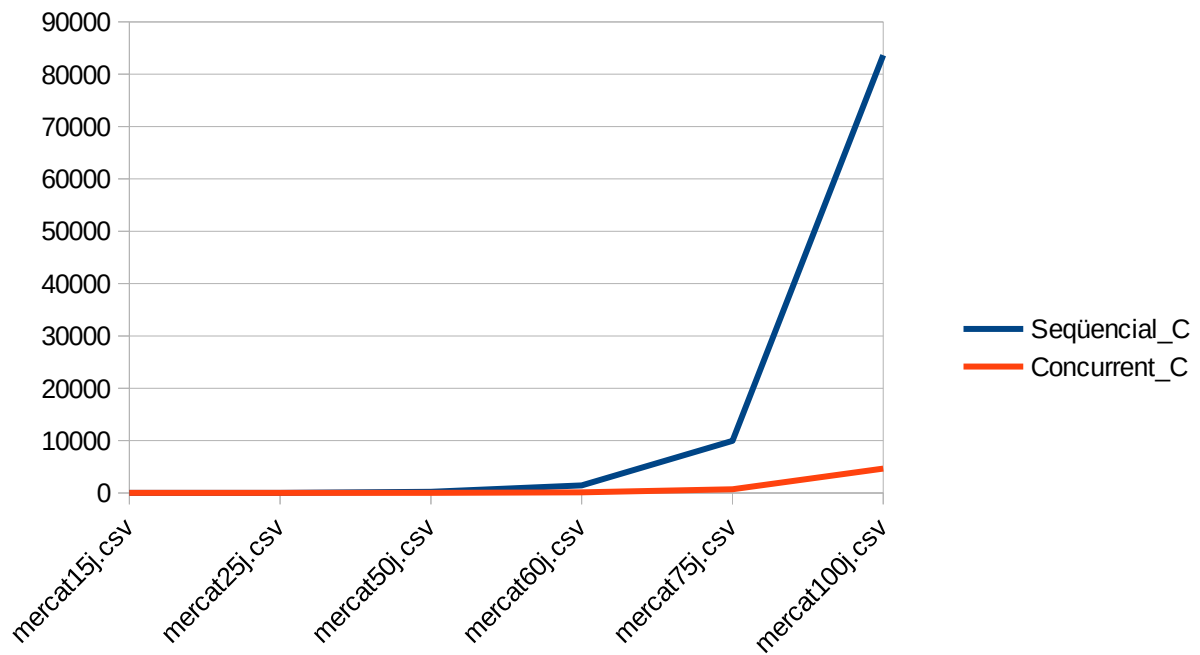
Fitxers	Temps			
	C		Java	
	Seqüencial_C	Concurrent_C	Seqüencial_Java	Concurrent_Java
mercat15j	0,198''	0,11''	0,202''	0,467''
mercat25j	4,119''	0,229''	0,8''	1,493''
mercat50j	3' 36,661''	11,234''	45,321''	24,048''
mercat60j	24' 14,709''	1' 44,693''	5' 12,662''	2' 29,493''
mercat75j	2h 45' 52,911''	11' 36,760''	3h 12' 34,865''	19' 56,374''
mercat100j	23h 13' 32,031''	1h 47' 29,649''	22h 28' 54,274''	2h 13' 41,548''

Després d'executar, per a cada fitxer, tan els programes seqüencials com els concurrents, es pot observar que al mercat15j.csv la millora de temps es molt poca per no dir quasi inexistent. En Java es pot observar que l'execució de forma concurrent tarda una mica més que la seqüencial, això es degut al sobrecost de la recursivitat.

A partir del fitxer mercat50j.csv ja es pot veure un notable increment en la millora del temps de forma concurrent.

Per als fitxers mercat60j.csv, mercat75j.csv i mercat100j.csv la diferència de temps es abismal, ja que la diferència entre l'execució seqüencial i concurrent pot arribar a ser de minuts i fins i tot d'hores degut a l'increment exponencial del número d'equips amb l'increment del número de jugadors.

Com a resultat d'aquest anàlisi podem destacar que l'aplicació de *threads* als programes fa que aquests tinguin un major rendiment i eficiència.



## Conclusions

Ha estat interessant implementar aquest problema d'optimització i observar l'efecte que comporta l'aplicació de *threads* pel que fa al rendiment com al decrement de la complexitat del problema.