

Universidad de Barcelona

Arthur Font Gouveia 20222613

Marc Colominas Casabella 20081692

## **Sistemes Operatius II**

### Pràctica 4

Barcelona

2021

# Índex

1. Introducció
2. Implementació
3. Conclusions

## 1. Introducció

En aquesta pràctica veurem com utilitzar la programació paral·lela per executar una funció de forma més ràpida aprofitant la capacitat multiprocessador.

En concret ens centrem en el paral·lisme per descomposició en les dades.

## 2. Implementació

### 2.1. Primera implementació

La primera part funciona correctament.

### 2.2 Segona Implementació

#### **Preguntes:**

- Quina és la secció crítica?

**R:** La secció crítica és accedir i actualitzar la variable global “last\_block”.

```
pthread_mutex_lock(&mutex);  
col_begin = last_block + 1;  
col_end = col_begin + N;  
if (col_end > recommendationMatrix->numberOfColumns) {  
    col_end = recommendationMatrix->numberOfColumns;  
}  
last_block = col_end;  
pthread_mutex_unlock(&mutex);
```

- Farà falta una secció crítica per realitzar el reduce?

**R:** No, perquè en la fas del reduce, es recullen els resultats parcials de cada fil i es calcula el resultat final fent servir variables locals.

#### **Experiments:**

1. Proveu d'executar l'aplicació amb  $M = 2$  i diferents valors d' $N$  ( $N \geq 1$ ) com, per exemple,  $N = 1, 10, 100, 1.000, 10.000$ . Dóna sempre el mateix resultat?

**R:** Sí, sempre ens dóna el resultat correcte.

2. Utilitzeu la comanda time per tal de mesurar el temps d'execució per als diferents valors d' $N$  proposats. Recordar que cal executar aquests experiments en

una màquina Linux (o Mac) nativa. Comenteu els resultats obtinguts i raoneu les diferències entre aquests. Hi ha valors d' $N$  que sigui més òptims, a nivell de temps d'execució, que altres valors?

**R:** Segueixen el temps d'execució retornat per cada valor d' $N$ .

Per  $N = 1$ :

real	0m45.091s
user	1m29.352s
sys	0m0.100s

Per  $N = 10$ :

real	0m44.850s
user	1m28.615s
sys	0m0.112s

Per  $N = 100$ :

real	0m45.689s
user	1m29.019s
sys	0m0.129s

Per  $N = 1.000$ :

real	0m49.582s
user	1m28.302s
sys	0m0.092s

Per  $N = 10.000$ :

real	1m22.444s
user	1m22.340s
sys	0m0.096s

Amb els resultats del time podem observar que el valor d' $N$  més òptim (per  $M=2$ ) és  $N = 10$ , ja que té el temps d'execució real més baix. Per  $N$  més grans podem observar que el temps d'execució real també són més grans. Per  $N = 10.000$ , el número és massa gran, aleshores el que passa és que només un fil executa tota la matriu, es pot veure mitjançant la suma del temps de usuari més el de temps de sistema, que és pràcticament igual al temps real.

### 3. Conclusions

Hem acabat la pràctica correctament i hem aconseguit el esperat, hem après com podem executar més ràpid una funció aprofitant la capacitat del multiprocessador amb la programació paral·lela.