



① a) versión paralela del bucle externo

```
double j (double A[N][N], double v[N]) {
    int i, j;
    double x, y, q, ming;
    #pragma omp parallel for reduction(+:y) private (ming, x, q)
    :
    // (a partir de aquí el código se mantendrá igual)
```

b) versión paralela del bucle interno

```
double j (double A[N][N], double v[N]) {
    int i, j;
    double x, y, q, ming;
    y = 0;
    for (j = 0; j < N; j++) {
        ming = 1e300;
        x = 1;
        #pragma omp parallel
        {
            #pragma omp for reduction(*:x) reduction(+:y)
            for {
                :
            }

            #pragma omp for private (q)
            for ( ... 1 ) {
                q = ...
                #pragma omp critical
                if (q < ming) {
                    ming = q;
                }
            }
        }
        // resto del código igual.
```


Miguel Ángel Navarro

7058868-M

17-11-2020

$$c) \sum_{j=0}^{N-1} \left(\underbrace{\sum_{i=j}^{N-1} (2)}_{2N-2j} + 3 + \underbrace{\sum_{i=0}^{N-1} (7)}_{7N} \right)$$

$$2N-2j + 3 + 7N$$

$$9N-2j+3$$

$$\left[\right]$$

$$8N^2 + 5N$$

← SECUENCIAL

$$\sum_{j=0}^{N-1} \left(\underbrace{\sum_{i=j}^{N-1} (2)}_{2N-2j} + 3 + \underbrace{\sum_{i=0}^{N-1} (7)}_{\frac{7N}{p}} \right)$$

$$2N-2j + 3 + \frac{7N}{p}$$

$$\frac{2Np+7N}{p} - 2j + 3$$

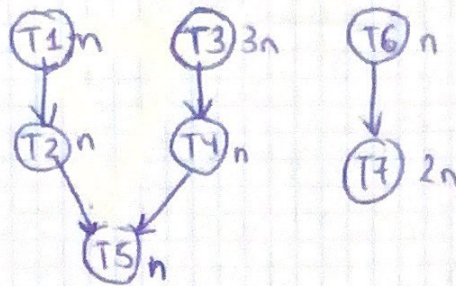
$$\left[\right]$$

$$\frac{n_p^2 + 7n^2 + 4pn}{p}$$

← PARALELO



2) a)



• Camino crítico $\rightarrow T3-T4-T5 \rightarrow \boxed{L=5n}$ \rightarrow longitud camino crítico

$$\rho_{mc} = \frac{\sum_{i=1}^N (\text{coste } a_i)}{L} = \frac{10n}{5n} = \boxed{2}$$

\rightarrow simb N el n° de tareas

b) Paralleliza

double / (double ya[], double yb[], double yc[], int n) {

double x;

#pragma omp parallel sections

{

#pragma omp section { /* section 1 */

x = min(ya, n);

inverte(ya, n);

#pragma omp section { /* section 2 */

normaliza(yb, n);

inverte(yb, n);

#pragma omp section { /* section 3 */

suma(ya, yb, n);

}

#pragma omp section { /* section 4 */

inverte(yc, n);

raiz(yc, n);

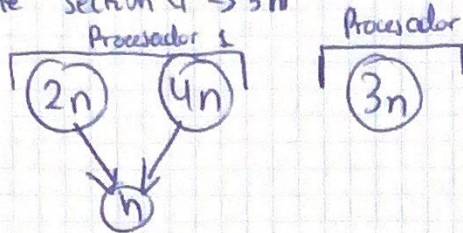
}

{ return x; /* podríamos ahorrarnos section 3 y sacar aquí T5 */

}

c) Speedup

- Coste section 1 $\rightarrow 2n$
- Coste section 2 $\rightarrow 4n$
- Coste section 3 $\rightarrow n$
- Coste section 4 $\rightarrow 3n$



$5n \rightarrow$ camino crítico (longitud)

$$\boxed{\text{Speedup} = \frac{10n}{5n} = 2}$$

Miguel Angel Navarro Arenas

70588808-14

17-11-2020

③ a) Paraleliza con 1 única región paralela

```
void gestiona_audienas (... ) {
```

```
    for (... ) {
```

```
        Suma Audienas ...
```

```
    {
```

```
        #pragma omp parallel {
```

```
            #pragma omp for reduction(+:total) reduction(+:audiencia) private (serie)
```

```
                for (i=0; ... ) {
```

```
                    #pragma omp atomic {
```

```
                        nCapitulos [serie] ++;
```

```
                    } /* total += audiencia; */
```

```
                } /* Cierra el for */
```

```
        #pragma omp for reduction(max:AudienciaMax) private (audiencia)
```

```
        for (i=0 ... ) {
```

```
            #pragma omp critical {
```

```
                if (audiencia > AudienciaMax) {
```

```
                    AudienciaMax = audiencia;
```

```
                    Mejor serie = i;
```

```
                }
```

```
                { /* cerramos sección crítica */
```

```
            } /* cerramos bucle for */
```

```
            /* sección de los printf */
```

```
        } /* cerramos función
```

int hilo = omp_get_thread_num();

nº hilos = 0;

nº hilos ++;

printf ("Hilo: %d, nº hilos: %d, hilo, nº hilos);

NOTA: en verde apartado b)