

Computación Paralela Y Distribuída 2020-II

Open Multiprocessing

Prof. J.Fiestas

Ejercicio 1: variables privadas

Escribir un programa que modifique dos variables privadas $i=1, j=2$ dentro de una directiva en paralelo, pero no cambie los valores de las variables originales.

Para ello, realice `assert(*ptr_i == 1, *ptr_j == 2)` dentro de la zona en paralelo, donde `ptr_i=&i`, y `ptr_j=&j`

Asimismo, realice un segundo `assert (i == 1 && j == 2)` fuera de la region en paralelo.

Imprima los valores de i y j dentro y fuera de la región en paralelo, y observe como cambian los valores dentro pero no fuera.

Utilize 4 hilos para OPENMP

Ejercicio 2: ordenamiento

Utilizar el constructor de sincronización **ORDERED** para ordenar secuencialmente los índices de un array de 100 elementos, escritos de 5 en 5 y empezando de 0. Para ello declare **...for ORDERED ...** antes del for loop que imprima los valores del array. No se olvide de declarar **... ORDERED** justo antes de la impresion.

Ejercicio 3: Variables compartidas/privadas y reducción

Reescribir el siguiente programa con solo las directivas OPENMP siguientes:

`#pragma omp parallel` , antes de inicializar `a=0`

`#pragma omp for reduction (+:a)`

Observar el resultado de la suma luego de varios intentos. ¿Se mantiene el resultado y por qué?

```
#include <iostream>
int main () {
    int a, i;
    a = 0;
    for (i = 0; i < 10; i++) {
        a += i;
    }
    Cout<<"La suma es: "
    <<a<<endl;
}
}
```

Ejercicio 4: Variables compartidas/privadas y reducción

Reescribir el programa con las directivas de **sincronización** OPENMP adecuadas, tal que:

a sea una variable compartida `shared(a)`, e

i sea una privada `private(i)`

La inicialización de **a** se haga solo por el
master thread

La impresión de la suma la haga solo un thread
(directiva `single`)

Compruebe si el resultado es ahora correcto.

En caso contrario, ¿Donde sería adecuado incluir una barrera y por qué? `#pragma omp barrier`

Escribir un programa similar para hallar el mínimo y el máximo de un array de 100 elementos generados aleatoriamente entre 1 y 99