

# Práctica de OpenMP 2

Raúl Sánchez Trujillo

21 de noviembre de 2021

# Índice general

<b>1. Introducción</b>	<b>1</b>
1.1. Primera versión.....	2
1.2. Segunda versión.....	3
<b>2. Conclusión</b>	<b>6</b>

# Capítulo 1

## Introducción

En esta práctica se van a realizar 2 versiones paralelizadas usando tasks en openmp sobre un código recursivo del cálculo de fibonacci.

Estas 2 versiones se diferenciarán en el uso de un tamaño mínimo a partir del cual se empezará a crear las tasks.

## 1.1. Primera versión

En la primera versión se observa al ejecutar el programa las siguientes salidas:

Comparando el iterativo con threads 32,16,8,4,2 respectivamente:

```
arqp009@yuca:~/PracticasARQP/practicaOMP2$ export OMP_NUM_THREADS=32
arqp009@yuca:~/PracticasARQP/practicaOMP2$ ./fibonacci-1 30
Compute Fibonacci number 30
Fibonacci number 30: 832040
Compute Time: 9.254691 seconds
arqp009@yuca:~/PracticasARQP/practicaOMP2$ ./fibonacci-2-8 30
Compute Fibonacci number 30
Fibonacci number 30: 832040
Compute Time: 0.325136 seconds
arqp009@yuca:~/PracticasARQP/practicaOMP2$ export OMP_NUM_THREADS=16
arqp009@yuca:~/PracticasARQP/practicaOMP2$ ./fibonacci-1 30
Compute Fibonacci number 30
Fibonacci number 30: 832040
Compute Time: 8.134914 seconds
arqp009@yuca:~/PracticasARQP/practicaOMP2$ ./fibonacci-2-8 30
Compute Fibonacci number 30
Fibonacci number 30: 832040
Compute Time: 0.267532 seconds
arqp009@yuca:~/PracticasARQP/practicaOMP2$ export OMP_NUM_THREADS=8
arqp009@yuca:~/PracticasARQP/practicaOMP2$ ./fibonacci-1 30
Compute Fibonacci number 30
Fibonacci number 30: 832040
Compute Time: 5.970785 seconds
arqp009@yuca:~/PracticasARQP/practicaOMP2$ ./fibonacci-2-8 30
Compute Fibonacci number 30
Fibonacci number 30: 832040
Compute Time: 0.185346 seconds
arqp009@yuca:~/PracticasARQP/practicaOMP2$ export OMP_NUM_THREADS=4
arqp009@yuca:~/PracticasARQP/practicaOMP2$ ./fibonacci-1 30
Compute Fibonacci number 30
Fibonacci number 30: 832040
Compute Time: 4.159837 seconds
arqp009@yuca:~/PracticasARQP/practicaOMP2$ ./fibonacci-2-8 30
Compute Fibonacci number 30
Fibonacci number 30: 832040
Compute Time: 0.125867 seconds
arqp009@yuca:~/PracticasARQP/practicaOMP2$ export OMP_NUM_THREADS=2
arqp009@yuca:~/PracticasARQP/practicaOMP2$ ./fibonacci-1 30
Compute Fibonacci number 30
Fibonacci number 30: 832040
Compute Time: 1.204496 seconds
arqp009@yuca:~/PracticasARQP/practicaOMP2$ ./fibonacci-2-8 30
Compute Fibonacci number 30
Fibonacci number 30: 832040
Compute Time: 0.139791 seconds
```

## 1.2. Segunda versión

En la segunda versión se observa al ejecutar el programa las siguientes salidas:

Comparando la primera versión frente a la segunda con threads 32,16,8,4,2 respectivamente y tamaño mínimo 8:

```
arqp009@yuca:~/PracticasARQP/practicaOMP2$ export OMP_NUM_THREADS=32
arqp009@yuca:~/PracticasARQP/practicaOMP2$ ./fibonacci-1 30
Compute Fibonacci number 30
Fibonacci number 30: 832040
Compute Time: 9.254691 seconds
arqp009@yuca:~/PracticasARQP/practicaOMP2$ ./fibonacci-2-8 30
Compute Fibonacci number 30
Fibonacci number 30: 832040
Compute Time: 0.325136 seconds
arqp009@yuca:~/PracticasARQP/practicaOMP2$ export OMP_NUM_THREADS=16
arqp009@yuca:~/PracticasARQP/practicaOMP2$ ./fibonacci-1 30
Compute Fibonacci number 30
Fibonacci number 30: 832040
Compute Time: 8.134914 seconds
arqp009@yuca:~/PracticasARQP/practicaOMP2$ ./fibonacci-2-8 30
Compute Fibonacci number 30
Fibonacci number 30: 832040
Compute Time: 0.267532 seconds
arqp009@yuca:~/PracticasARQP/practicaOMP2$ export OMP_NUM_THREADS=8
arqp009@yuca:~/PracticasARQP/practicaOMP2$ ./fibonacci-1 30
Compute Fibonacci number 30
Fibonacci number 30: 832040
Compute Time: 5.970785 seconds
arqp009@yuca:~/PracticasARQP/practicaOMP2$ ./fibonacci-2-8 30
Compute Fibonacci number 30
Fibonacci number 30: 832040
Compute Time: 0.185346 seconds
arqp009@yuca:~/PracticasARQP/practicaOMP2$ export OMP_NUM_THREADS=4
arqp009@yuca:~/PracticasARQP/practicaOMP2$ ./fibonacci-1 30
Compute Fibonacci number 30
Fibonacci number 30: 832040
Compute Time: 4.159837 seconds
arqp009@yuca:~/PracticasARQP/practicaOMP2$ ./fibonacci-2-8 30
Compute Fibonacci number 30
Fibonacci number 30: 832040
Compute Time: 0.125867 seconds
arqp009@yuca:~/PracticasARQP/practicaOMP2$ export OMP_NUM_THREADS=2
arqp009@yuca:~/PracticasARQP/practicaOMP2$ ./fibonacci-1 30
Compute Fibonacci number 30
Fibonacci number 30: 832040
Compute Time: 1.204496 seconds
arqp009@yuca:~/PracticasARQP/practicaOMP2$ ./fibonacci-2-8 30
Compute Fibonacci number 30
Fibonacci number 30: 832040
Compute Time: 0.139791 seconds
```



Comparando la primera versión frente a la segunda con threads 32,16,8,4,2 respectivamente y tamaño mínimo 16:

```
arqp009@yuca:~/PracticasARQP/practicaOMP2$ export OMP_NUM_THREADS=32
arqp009@yuca:~/PracticasARQP/practicaOMP2$ ./fibonacci-1 30
Compute Fibonacci number 30
Fibonacci number 30: 832040
Compute Time: 9.403602 seconds
arqp009@yuca:~/PracticasARQP/practicaOMP2$ ./fibonacci-2-16 30
Compute Fibonacci number 30
Fibonacci number 30: 832040
Compute Time: 0.010948 seconds
arqp009@yuca:~/PracticasARQP/practicaOMP2$ export OMP_NUM_THREADS=16
arqp009@yuca:~/PracticasARQP/practicaOMP2$ ./fibonacci-1 30
Compute Fibonacci number 30
Fibonacci number 30: 832040
Compute Time: 7.750581 seconds
arqp009@yuca:~/PracticasARQP/practicaOMP2$ ./fibonacci-2-16 30
Compute Fibonacci number 30
Fibonacci number 30: 832040
Compute Time: 0.007429 seconds
arqp009@yuca:~/PracticasARQP/practicaOMP2$ export OMP_NUM_THREADS=8
arqp009@yuca:~/PracticasARQP/practicaOMP2$ ./fibonacci-1 30
Compute Fibonacci number 30
Fibonacci number 30: 832040
Compute Time: 5.187445 seconds
arqp009@yuca:~/PracticasARQP/practicaOMP2$ ./fibonacci-2-16 30
Compute Fibonacci number 30
Fibonacci number 30: 832040
Compute Time: 0.008526 seconds
arqp009@yuca:~/PracticasARQP/practicaOMP2$ export OMP_NUM_THREADS=4
arqp009@yuca:~/PracticasARQP/practicaOMP2$ ./fibonacci-1 30
Compute Fibonacci number 30
Fibonacci number 30: 832040
Compute Time: 2.759758 seconds
arqp009@yuca:~/PracticasARQP/practicaOMP2$ ./fibonacci-2-16 30
Compute Fibonacci number 30
Fibonacci number 30: 832040
Compute Time: 0.011086 seconds
arqp009@yuca:~/PracticasARQP/practicaOMP2$ export OMP_NUM_THREADS=2
arqp009@yuca:~/PracticasARQP/practicaOMP2$ ./fibonacci-1 30
Compute Fibonacci number 30
Fibonacci number 30: 832040
Compute Time: 3.053844 seconds
arqp009@yuca:~/PracticasARQP/practicaOMP2$ ./fibonacci-2-16 30
Compute Fibonacci number 30
Fibonacci number 30: 832040
Compute Time: 0.015150 seconds
```

Comparando la primera versión frente a la segunda con threads 32,16,8,4,2 respectivamente y tamaño mínimo 32:

```
arqp009@yuca:~/PracticasARQP/practicaOMP2$ export OMP_NUM_THREADS=32
arqp009@yuca:~/PracticasARQP/practicaOMP2$ ./fibonacci-1 30
Compute Fibonacci number 30
Fibonacci number 30: 832040
Compute Time: 9.797748 seconds
arqp009@yuca:~/PracticasARQP/practicaOMP2$ ./fibonacci-2-32 30
Compute Fibonacci number 30
Fibonacci number 30: 832040
Compute Time: 0.020165 seconds
arqp009@yuca:~/PracticasARQP/practicaOMP2$ export OMP_NUM_THREADS=16
arqp009@yuca:~/PracticasARQP/practicaOMP2$ ./fibonacci-1 30
Compute Fibonacci number 30
Fibonacci number 30: 832040
Compute Time: 7.776085 seconds
arqp009@yuca:~/PracticasARQP/practicaOMP2$ ./fibonacci-2-32 30
Compute Fibonacci number 30
Fibonacci number 30: 832040
Compute Time: 0.021149 seconds
arqp009@yuca:~/PracticasARQP/practicaOMP2$ export OMP_NUM_THREADS=8
arqp009@yuca:~/PracticasARQP/practicaOMP2$ ./fibonacci-1 30
Compute Fibonacci number 30
Fibonacci number 30: 832040
Compute Time: 5.292701 seconds
arqp009@yuca:~/PracticasARQP/practicaOMP2$ ./fibonacci-2-32 30
Compute Fibonacci number 30
Fibonacci number 30: 832040
Compute Time: 0.019907 seconds
arqp009@yuca:~/PracticasARQP/practicaOMP2$ export OMP_NUM_THREADS=4
arqp009@yuca:~/PracticasARQP/practicaOMP2$ ./fibonacci-1 30
Compute Fibonacci number 30
Fibonacci number 30: 832040
Compute Time: 4.715260 seconds
arqp009@yuca:~/PracticasARQP/practicaOMP2$ ./fibonacci-2-32 30
Compute Fibonacci number 30
Fibonacci number 30: 832040
Compute Time: 0.020874 seconds
arqp009@yuca:~/PracticasARQP/practicaOMP2$ export OMP_NUM_THREADS=2
arqp009@yuca:~/PracticasARQP/practicaOMP2$ ./fibonacci-1 30
Compute Fibonacci number 30
Fibonacci number 30: 832040
Compute Time: 1.241567 seconds
arqp009@yuca:~/PracticasARQP/practicaOMP2$ ./fibonacci-2-32 30
Compute Fibonacci number 30
Fibonacci number 30: 832040
Compute Time: 0.017181 seconds
```

## Capítulo 2

### Conclusión

En conclusión, en la primera versión del código se observa como la versión iterativa es mucho mejor en cuestión de rendimiento, esto es debido a que, al hacer paralelismo de tasks a este algoritmo recursivo, no sirve de nada, ya que, debe esperar a que se calculen los 2 tasks anteriores para poder seguir con el cálculo por lo que se termina calculando de forma iterativa y no paralela. Además empeora debido a que la creación y gestión de los tasks produce un overhead que provoca un empeoramiento increíble frente al no paralelizado, por eso cuantos menos threads usemos mejor será el rendimiento del mismo, ya que, tendrá que gestionar un menor número de threads. En la segunda versión se consigue mejorar la primera versión, ya que, solo se paraleliza a partir de un cierto número mínimo de  $N$ , por lo que, no tiene que crear tantos threads ni gestionarlos, y esto provoca que el overhead sea mucho menor y por lo tanto mejore el rendimiento con respecto a la primera versión.