# Hoja de trabajo semana 7

Nombre: Marlon Fuentes

Carné: 15240

**Competencias a desarrollar**

Comprende los fundamentos de computación paralela, distribuida, tipos de paralelismo y sus efectos en la programación, por medio de ejercicios prácticos . Usar directivas de OpenMP para realizar las siguientes tareas.

**Instrucciones**

Esta actividad se realizará de forma individual. Ejecute cada una de las actividades y anote sus resultados. Utilice la reflexión para resumir sus observaciones y discutir conclusiones de forma individual.

***Actividades***

1. **(5 minutos)** Utilice el arreglo mostrado al final del enunciado y ordénelo de forma ascendente utilizando el algoritmo **Trasposición Impar-Par**. Asegúrese de colocar los resultados para cada una de las rondas en la casilla correspondiente. Verifique que se necesitan *N = número de elementos* para poder ordenar el arreglo usando este algoritmo.

**A = [ D, A, C, E, U, N, B, M, T, F]**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **A0** | **A1** | **A2** | **A3** | **A4** | **A5** | **A6** | **A7** | **A8** | **A9** |
|  | **D** | **A** | **C** | **E** | **U** | **N** | **B** | **M** | **T** | **F** |
| **0** | **A** | **D** | **C** | **E** | **N** | **U** | **B** | **M** | **F** | **T** |
| **1** | **A** | **C** | **D** | **E** | **N** | **B** | **U** | **F** | **M** | **T** |
| **2** | **A** | **C** | **D** | **E** | **B** | **N** | **F** | **U** | **M** | **T** |
| **3** | **A** | **C** | **D** | **B** | **E** | **F** | **N** | **M** | **U** | **T** |
| **4** | **A** | **C** | **B** | **D** | **E** | **F** | **M** | **N** | **T** | **U** |
| **5** | **A** | **B** | **C** | **D** | **E** | **F** | **M** | **N** | **T** | **U** |
| **6** | **A** | **B** | **C** | **D** | **E** | **F** | **M** | **N** | **T** | **U** |

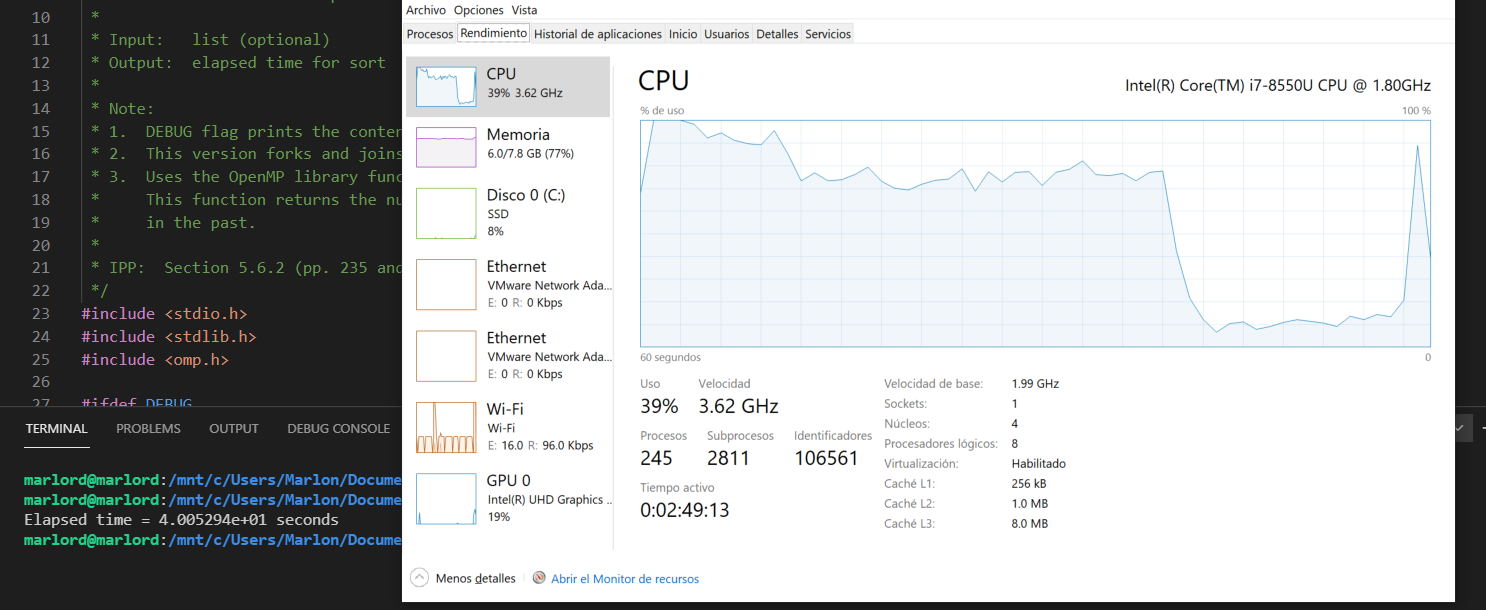
1. **(3 minutos)** Usando el arreglo original, ejecute la fase 0 y la fase 1 al mismo tiempo. Esto significa que para la fase 1 no debe usar los elementos traspuestos de la fase 0 sino los elementos originales. Ejecute de ahí en adelante cada fase de forma secuencial hasta la fase 3. Esto ilustra el riesgo de dependencia que tiene este algoritmo de clasificación.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **A0** | **A1** | **A2** | **A3** | **A4** | **A5** | **A6** | **A7** | **A8** | **A9** |
|  | **D** | **A** | **C** | **E** | **U** | **N** | **B** | **M** | **T** | **F** |
| **0** | **A** | **D** | **C** | **E** | **N** | **U** | **B** | **M** | **F** | **T** |
| **1** | **D** | **A** | **C** | **E** | **U** | **B** | **N** | **M** | **T** | **F** |
| **2** | **A** | **D** | **C** | **E** | **B** | **U** | **M** | **N** | **F** | **T** |
| **3** | **A** | **C** | **D** | **B** | **E** | **M** | **U** | **F** | **N** | **T** |
| **4** | **A** | **C** | **B** | **D** | **E** | **M** | **F** | **U** | **N** | **T** |
| **5** | **A** | **B** | **C** | **D** | **R** | **F** | **M** | **N** | **U** | **T** |
| **6** | **A** | **B** | **C** | **D** | **E** | **F** | **M** | **N** | **T** | **U** |

1. **(8 minutos)** Ejecute los programas omp\_odd\_even1.c. y omp\_odd\_even2.c para 200,000 elementos. Los argumentos del programa son: <HILOS> <ELEMENTOS> g. Abra su administrador de tareas para poder visualizar el uso de CPU y memoria. Compare la diferencia en tiempos de ejecución y recursos de cada versión. Ejecute al menos 3 veces cada uno para obtener un promedio. ¿Indican sus resultados que es mejor reutilizar un equipo de threads en lugar de crear y destruir equipos nuevos en cada iteración?

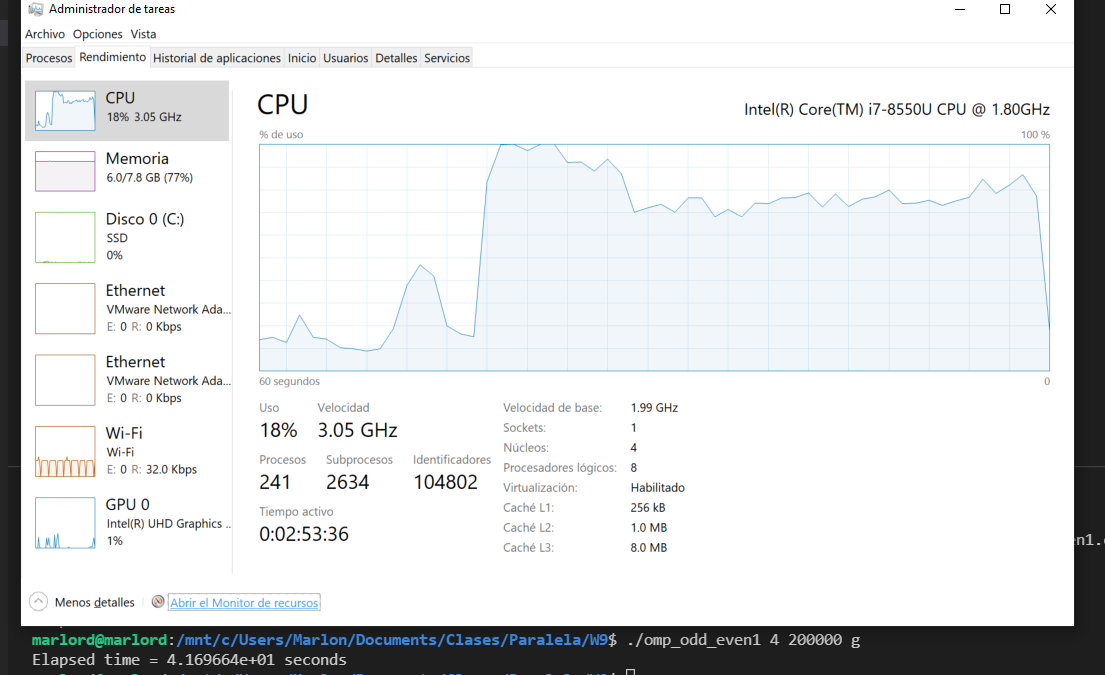
Omp\_odd\_even1

Iteraciön 1:

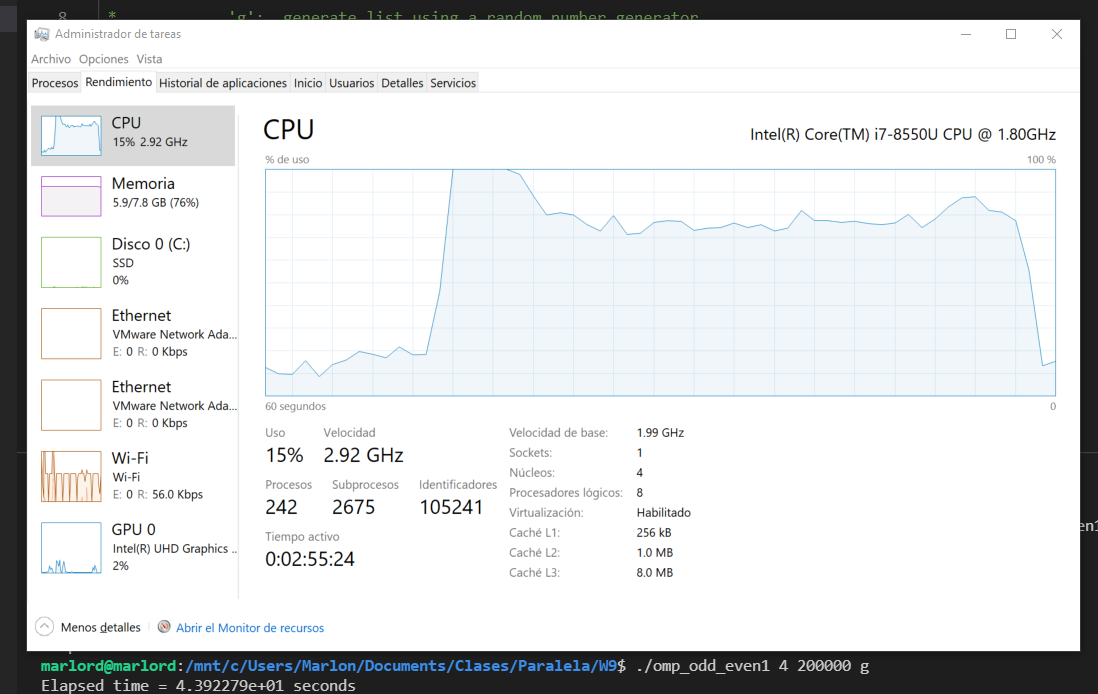


Resultado: 4.005294e+01 seconds

Iteración 2:



Resultado: 4.169664e+01

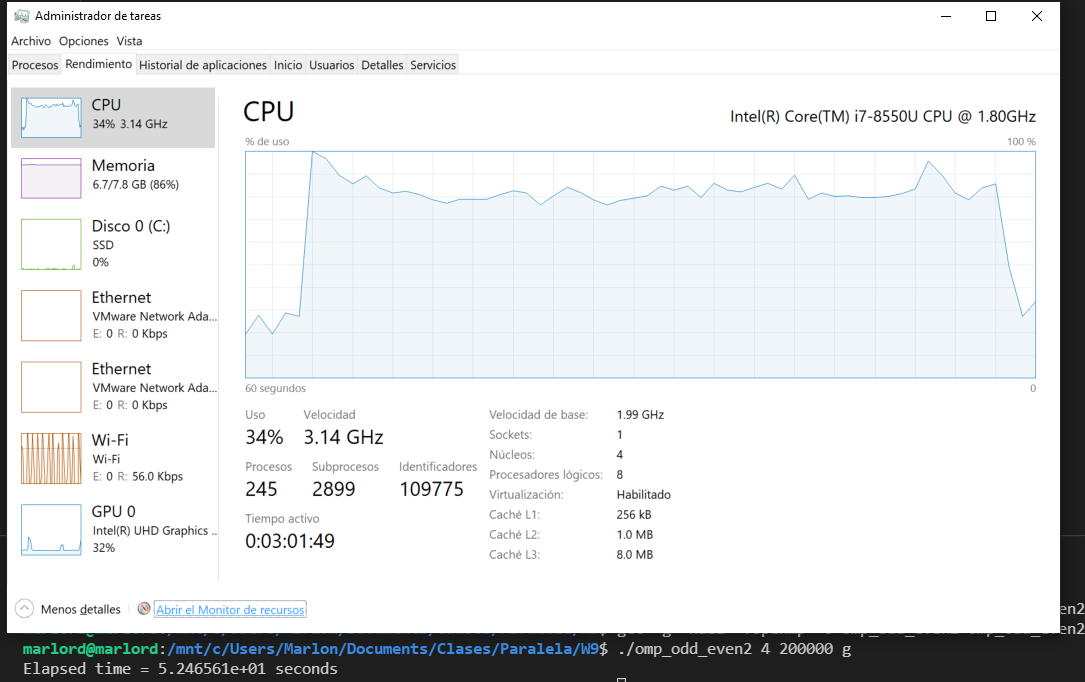


Resultado: 4.392279e+01

Promedio: 41.901

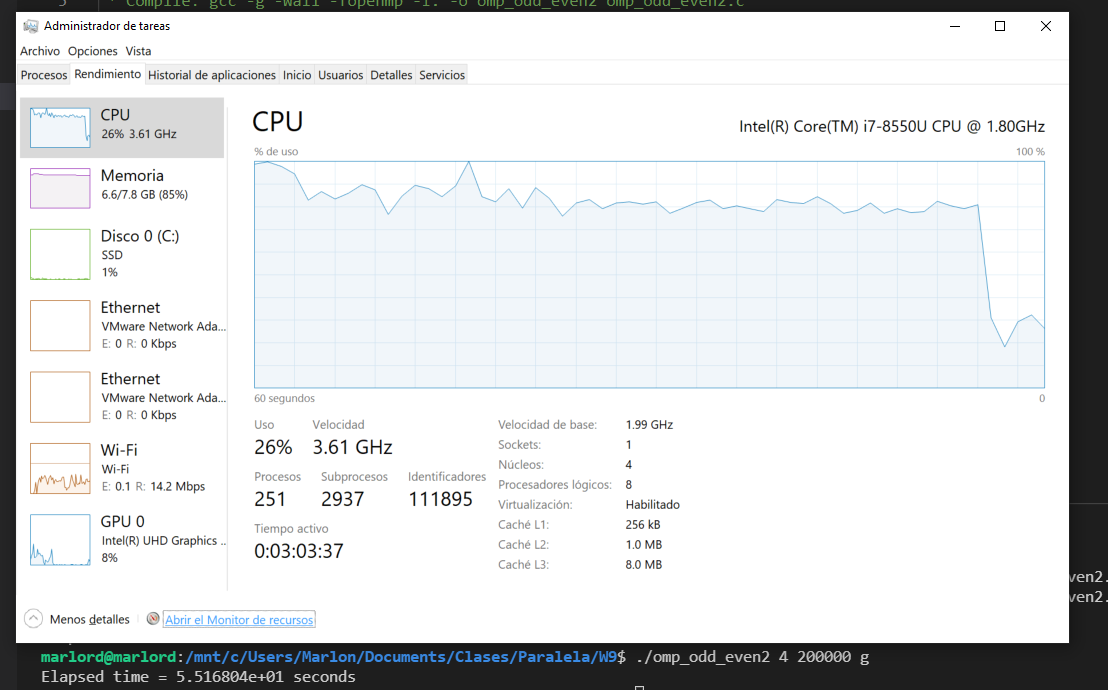
Omp\_odd\_even2

Iteración 1:



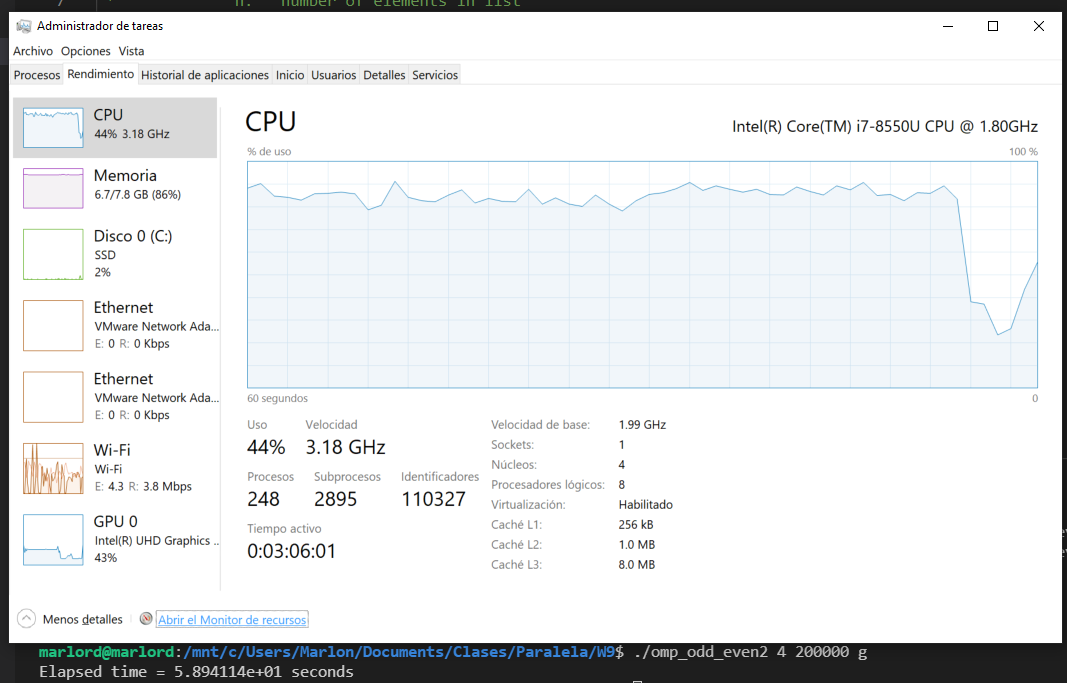
Resultado: 5.246561e+01

Iteración 2



Resultado: 5.516804e+01

Iteración 3



Resultado: 5.894114e+01

Promedio: 55.525

La reutilización resulto mucho mejor debido a que su promedio de tiempo es mejor que a la creación de nuevos. Esto muestra la eficiencia que es para los núcleos que es reutilizar recursos.

1. **(5 minutos)** Asuma que tiene un arreglo de 20 elementos. Liste la asignación de elementos a cada hilo, según el tipo de programación (schedule) e hilos indicada.
   1. schedule (static, 4); 5 hilos

|  |  |
| --- | --- |
| Thrhead | Hilos |
| 0 | 0,1,2,3,20,21,22,23,40,41,42,43 |
| 1 | 4,5,6,7,23,25,27,44,45,46,47 |
| 2 | 8,9,10,11,28,29,30,31,48,49,50 |
| 3 | 12,13,14,15,32,33,34,35 |
| 4 | 16,17,18,19,36,37,38,39 |

* 1. schedule (static, 2); 5 hilos

|  |  |
| --- | --- |
| Thread | Hilos |
| 0 | 0,1,10,11,20,21,30,31,40,41,49,50 |
| 1 | 2,3,12,13,22,23,32,33,42,43 |
| 2 | 4,5,14,15,24,25,34,35,44,45 |
| 3 | 6,7,16,17,28,36,37,45,46 |
| 4 | 8,9,18,29,28,29,38,39,47,48 |

* 1. schedule (static, 3); 4 hilos

1. **(8 minutos)** Usando un diagrama de tiempo, ilustre la asignación de un arreglo de 50 elementos de un schedule (dynamic). Asuma que tienen 4 hilos, T0 y T3 tardan 2 ciclos, T1 tarda 3 ciclos y T2 tarda 4 ciclos en completar su chunk. Realice el diagrama para un chunksize = 3.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ciclo | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| T0 | A0  A1  A2 | A0  A1  A2 | A12  A13  A14 | A12  A13  A14 | A21  A22  A23 | A21  A22  A23 | A27  A28  A29 | A36  A37  A38 | A36  A37  A38 | A36  A37  A38 |  |  |  |  |  |  |
| T1 | A3  A4  A5 | A3  A4  A5 | A3  A4  A5 | A18  A19  A20 | A18  A19  A20 | A18  A19  A20 | A30  A31  A32 | A30  A31  A32 | A30  A31  A32 | A45  A46  A47 | A45  A46  A47 | A45  A46  A47 |  |  |  |  |
| T2 | A6  A7  A8 | A6  A7  A8 | A6  A7  A8 | A6  A7  A8 | A24  A25  A26 | A24  A25  A26 | A24  A25  A26 | A39  A40  A41 | A39  A40  A41 | A39  A40  A41 | A39  A40  A41 | A39  A40  A41 |  |  |  |  |
| T3 | A9  A10  A11 | A9  A10  A11 | A15  A16  A17 | A15  A16  A17 | A27  A28  A29 | A27  A28  A29 | A33  A34  A35 | A33  A34  A35 | A42  A43  A44 | A42  A43  A44 | A48  A49  A50 | A48  A49  A50 |  |  |  |  |