Instituto Tecnológico de Costa Rica



Escuela de Computación

Principio de Sistemas Operativos

Jorge Alfaro Velazco

Daniel Berrocal Ramírez

201049486

Jorge Rojas Aragonés

200969830

San Carlos, 6 de Noviembre del 2013

Contenido

[Descripción: 3](#_Toc371513055)

[Solución. 3](#_Toc371513056)

[Problemas en el Desarrollo 4](#_Toc371513057)

[Resultados Obtenidos 4](#_Toc371513058)

[Estados de Ejecución 4](#_Toc371513059)

[Estado 1: No Ejecución. 5](#_Toc371513060)

[Estado 2: Ejecución Secuencial 7](#_Toc371513061)

[Estado 3: Ejecución Paralela 9](#_Toc371513062)

[Comparación entre los 3 estados 11](#_Toc371513063)

[Conclusiones: 12](#_Toc371513064)

[Bibliografía: 13](#_Toc371513065)

# Descripción:

Hoy en día el consumo y procesamiento de datos relativamente grandes es uno de los puntos más relevantes en el campo de la computación actual, las grandes compañías que manejan cantidades exuberantes de datos ya no pueden consumir o procesar estos datos de la forma habitual en que se hace, es ahí donde surge la programación paralela donde se aprovecha al máximo las bondades que de los procesadores multinucleo, permitiendo ejecutar varios procesos a la vez, uno o varios procesos sobre cada núcleo del procesador.

El proyecto se basa en la elaboración de una aplicación que permita resolver problemas de manera secuencial y/o paralela y comparar los tiempos de ejecución entre las 2 maneras de ejecución.

# Solución.

Para llevar a cabo la elaboración de este proyecto, se pasó por varias etapas de desarrollo, esto para no recargar la labor a realizar en un solo punto, en conjunto se decidió dividir el proyecto en varias etapas, pensando en un mejor resultado al final, cada etapa se realizó en un tiempo considerable dependiendo de la etapa que se tratase, a continuación se describe cada una de las etapas:

* Investigación:

En esta etapa se realizó una intensa búsqueda de todo tipo de información que proporciona algún apoyo para la confección del sistema.

Entre los temas más importantes que se investigaron se encuentran:

* Métodos de encriptación.
* Métodos de ordenamiento.
* Programación paralela.
* Entre otros.

* Planteo de la Solución:

En dicha etapa, con toda la información que se recaudó en la etapa de investigación, se procede a plantear posibles maneras de llegar a una solución, discutiendo, probando, y elaborando pequeñas pruebas de código, mutuamente se llegó a la decisión de cuál posible solución implementar.

* Desarrollo:

Teniendo en mente de qué manera darle solución al problema, se inicia con el desarrollo de la solución, en esta etapa se implementa todo el conocimiento adquirido en clase y  la información recaudada en la etapa de investigación.

* Prueba:

En esta etapa, se le realizaron distintas pruebas al sistema para buscar todo tipo de errores y generar las posibles soluciones para dichos errores.

# Problemas en el Desarrollo

Durante la etapa de desarrollo se presentaron algunos inconvenientes que dieron como resultado un gasto de tiempo considerable. Entre los principales inconvenientes se encuentran:

* Al tener que trabar con datos relativamente grandes, el consumo de memoria por parte de la aplicación daba como resultado que el equipo presentara problemas de ejecución, esto se puede resumir en que al no tener equipos adecuados o con las condiciones necesarias para la ejecución y consumo de datos tan grandes se presenta un retraso considerable en el tiempo de desarrollo.
* Al no tener mucho conocimiento sobre la implementación de algoritmos paralelos, esto provoco que se re implementaran algunos algoritmos ya hechos, esto porque al ejecutarse de manera paralela se consumían más recursos de los que se debían.

# Resultados Obtenidos

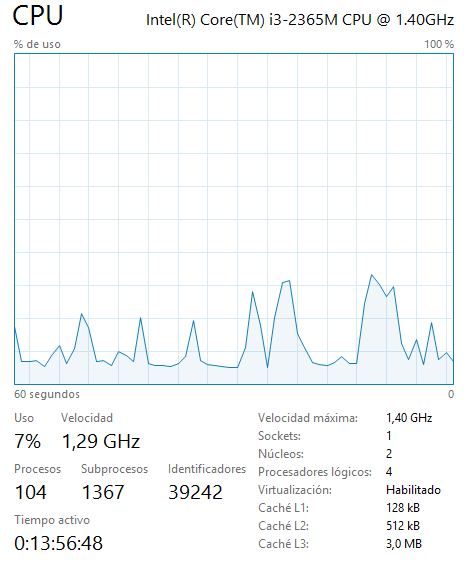
Es importante mencionar que al comparar los tiempos de ejecución se logró obtener una reducción considerable entre el tiempo de ejecución secuencial y el paralelo, por ejemplo encriptando un libro de más de 200 páginas los tiempos de ejecución fueron:

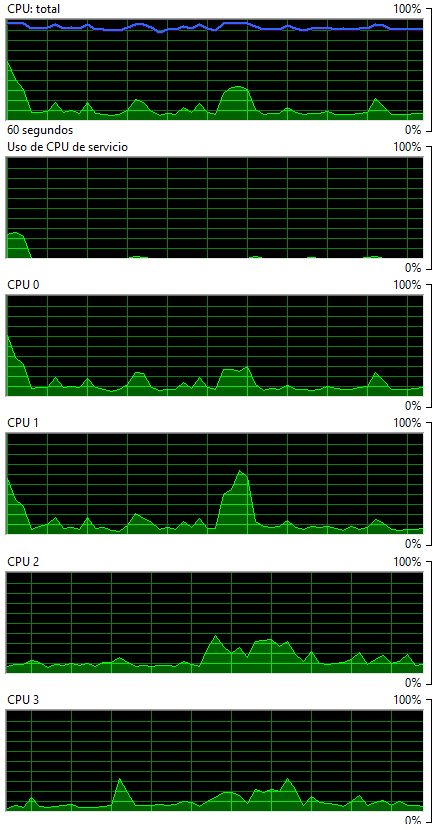
* Secuencial: los tiempos siempre fueron mayores a los cuatro minutos y medio.
* Paralelo: los tiempos variaron entre un minuto y medio a los dos minutos y medio.

# Estados de Ejecución

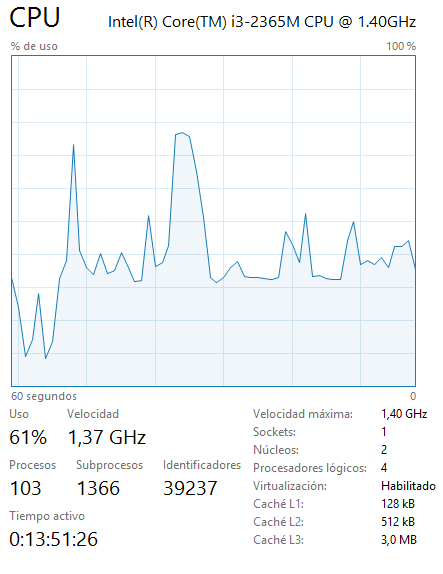
A continuación se mostrara los diferentes niveles de recursos consumidos por la CPU en los 3 diferentes estados de ejecución (no ejecutando, ejecutando paralelo y ejecutando secuencial).

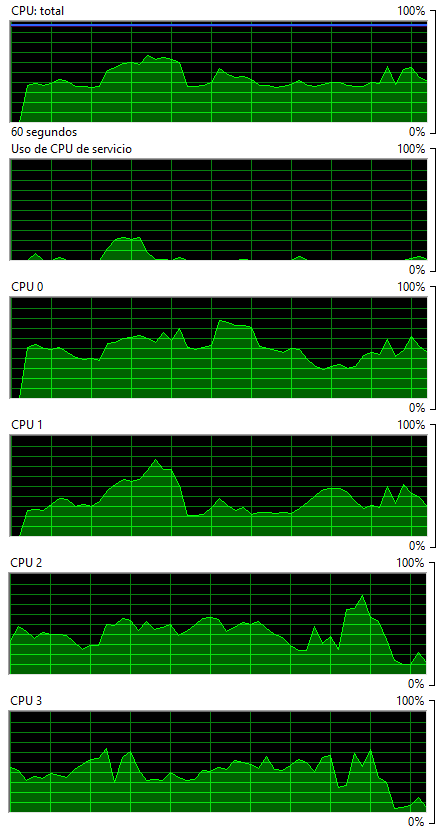
## Estado 1: No Ejecución.



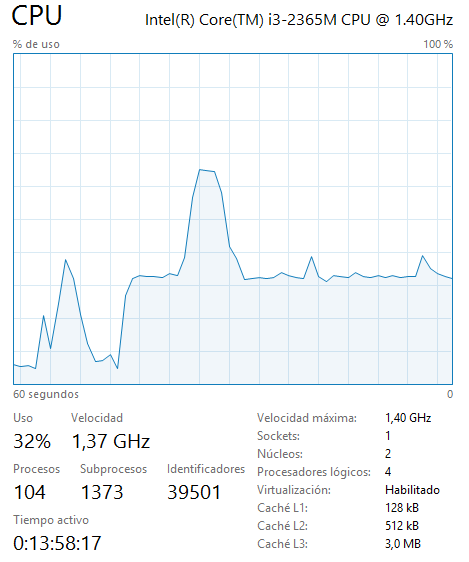


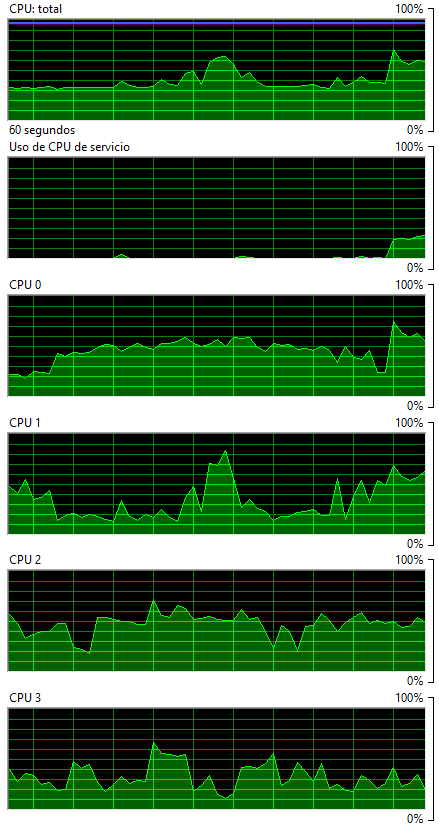
## Estado 2: Ejecución Secuencial



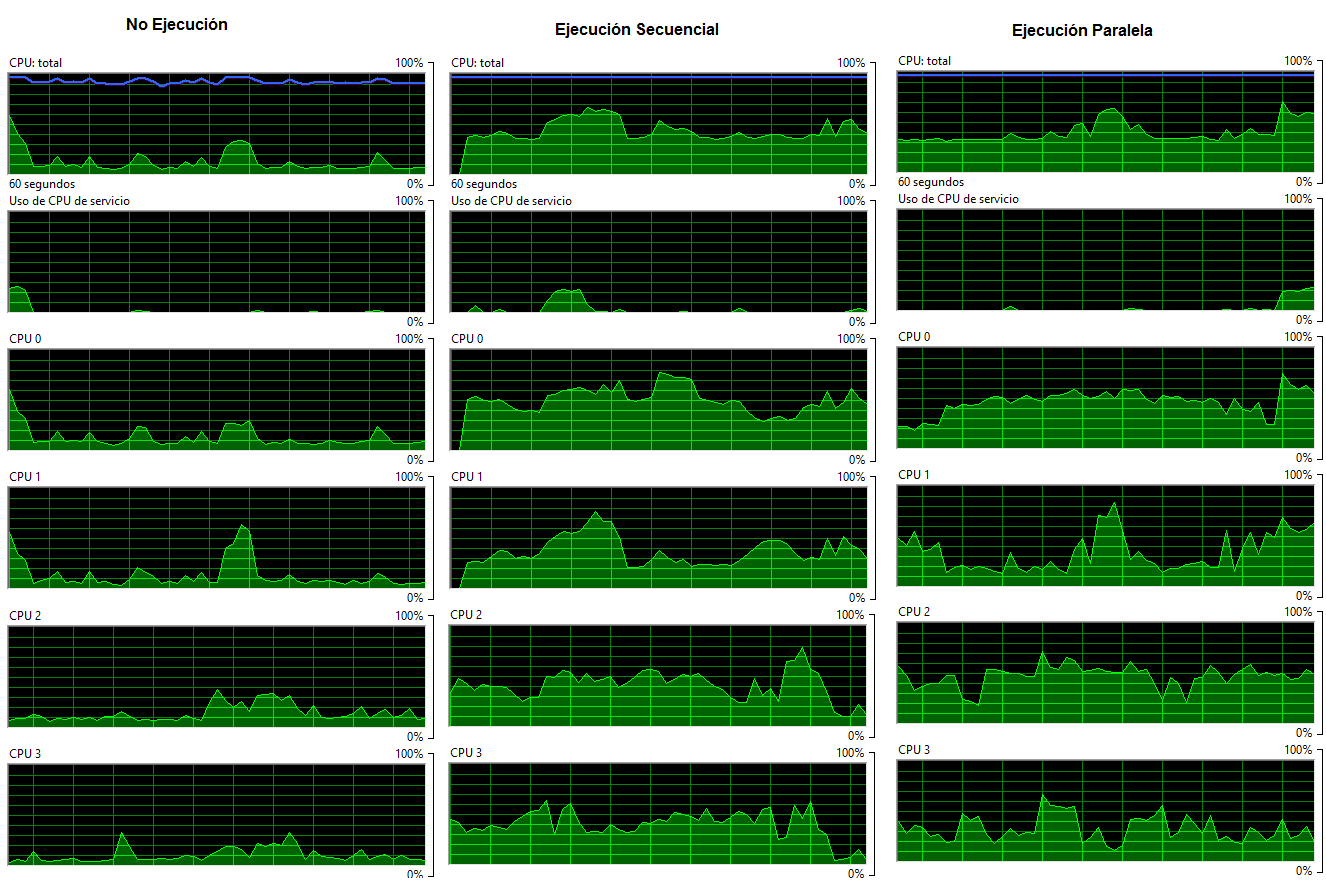


## Estado 3: Ejecución Paralela

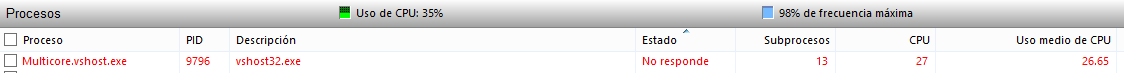




## Comparación entre los 3 estados



Como se puede ver, en algunas ocasiones el equipo no respondía a como se desearía ya que este no cuenta con un gran procesador que ejecute con facilidad el consumo de los recursos necesarios.



# Conclusiones:

Durante el desarrollo de este proyecto, se logran rescatar muchos aspectos importantes para cualquier estudiante de la carrera de ingeniería en computación, pero el que más importancia parece tener es, el impacto a nivel académico, que se da cuando se analizan los procesos que se llevan a cabo cuando se tiene que consumir datos relativamente grandes, las diferentes adaptaciones que se le deben de realizar a un algoritmo para implementarlo de forma paralela, los diferentes niveles de consumo de recursos entre los procesos que se ejecutan secuencial y paralelamente. Por esta razón es importante la asignación de este tipo de proyectos en los cuales se inculca una manera de pensar analítica, ayudando esto a la formación de futuros ingenieros profesionales.

# Bibliografía:

Cifrado César. (n.d.). Wikipedia. Retrieved November 6, 2013, from http://es.wikipedia.org/wiki/Cifrado\_C%C3%A9sar

Cifrado XOR. (n.d.). - Wikipedia, la enciclopedia libre. Retrieved November 6, 2013, from http://es.wikipedia.org/wiki/Cifrado\_XOR

Cifrado XOR. (n.d.). Cifrado XOR. Retrieved November 6, 2013, from http://fabbc.awardspace.com/Cifrado-XOR.php

Como medir tiempo de ejecucion de codigo en C# y .NET. (n.d.). Gabriel Rodriguez Plancarte. Retrieved November 6, 2013, from http://gabrielrodriguez.net/como-medir-tiempo-

Encriptación Simétrica con C#.NET. (n.d.). El blog del burgus. Retrieved November 6, 2013, from http://elburgues.wordpress.com/2010/03/13/encriptacion-simetrica-en-c/

Merge sort. (2013, February 11). Wikipedia. Retrieved November 6, 2013, from http://en.wikipedia.org/wiki/Merge\_sort

Quicksort. (n.d.). - Wikipedia, la enciclopedia libre. Retrieved November 6, 2013, from <http://es.wikipedia.org/wiki/Quicksort>

Parallel Class. (n.d.). *(System.Threading.Tasks)*. Retrieved November 6, 2013, from http://msdn.microsoft.com/en-us/library/system.threading.tasks.parallel(v=vs.110).aspx