



Análisis de complejidad y correctitud de los algoritmos de compresión y descompresión del sistema de empaquetado lsh(Implementación LZW-Hash)

Estudiantes:

Willy Villalobos Marrero B17170

Daniel Méndez Zeledón A83911

Javier Acosta Villalobos A80056

1. Introducción

A partir de la segunda mitad del curso y como complemento de la programación en C++, se aprendió a realizar análisis que permiten determinar si los algoritmos son óptimos, que tan rápidos son y como mejorarlos. Como parte del proyecto, se va a tomar el algoritmo descrito por uno de los presentes en este proyecto como parte del curso y se le harán mejoras en caso de ser necesario, además de analizar su complejidad y su correctitud. El algoritmo que se va a analizar es el LZW utilizando tabla Hash, el cual genera un gran aporte a la compresión de los archivos.

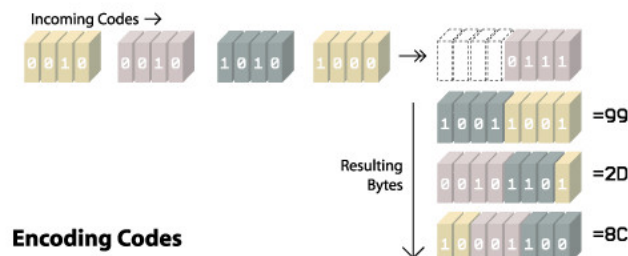


Figura 1: Muestra de compresión LZW



También, como parte del aporte al trabajo previo, se creará el algoritmo que descomprima los archivos de formato ilsh creados por el compresor y también se tratará de utilizar la teoría para hacer que éste sea lo más eficiente posible. De igual manera se le harán las respectivas pruebas de complejidad y correctitud paso por paso para así identificar fallas del código como tal y las posibles mejoras a éstos.

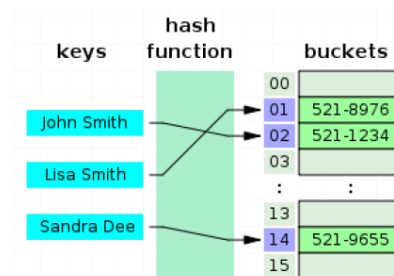


Figura 2: Tabla Hash

2. Objetivos

2.1. Objetivo General

Analizar el algoritmo para la implementación del LZW con tabla Hash con el fin de obtener su rendimiento a partir del análisis de correctitud y complejidad aprendidos en el curso para determinar si existe mejoras frente a otros compresores de archivos.

2.2. Objetivos Específicos

- Analizar la complejidad del algoritmo de compresión de LZW con tabla Hash para determinar el desempeño del algoritmo, utilizando un análisis línea a línea de la cantidad de operaciones que hace éste.
- Realizar un análisis de correctitud con el fin de demostrar que el algoritmo es correcto, utilizando los invariantes de lazo que se puedan determinar para tal demostración.
- Crear el algoritmo para el descompresor que no se había tenido listo y a su vez analizar su complejidad y correctitud de la misma manera que para el compresor, de las formas previamente descritas.
- Crear los esquemas del modelo de ejecución del algoritmo con el fin de tener una mejor representación del algoritmo y así mejorar su documentación.

3. Metodología

Se hará un pequeño análisis previo, se verán las posibilidades de analizar eficazmente el algoritmo de compresión de LZW, una vez hecho esto, se hará un planeamiento de las tareas a realizar por cada uno de los miembros en conjunto, para acelerar el proceso de elaboración del documento final. Se comenzará por hacer el análisis de complejidad del algoritmo de



UNIVERSIDAD DE COSTA RICA
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA ELÉCTRICA
**ESTRUCTURAS ABSTRACTAS DE DATOS Y
ALGORITMOS PARA INGENIERÍA**

IE-0217

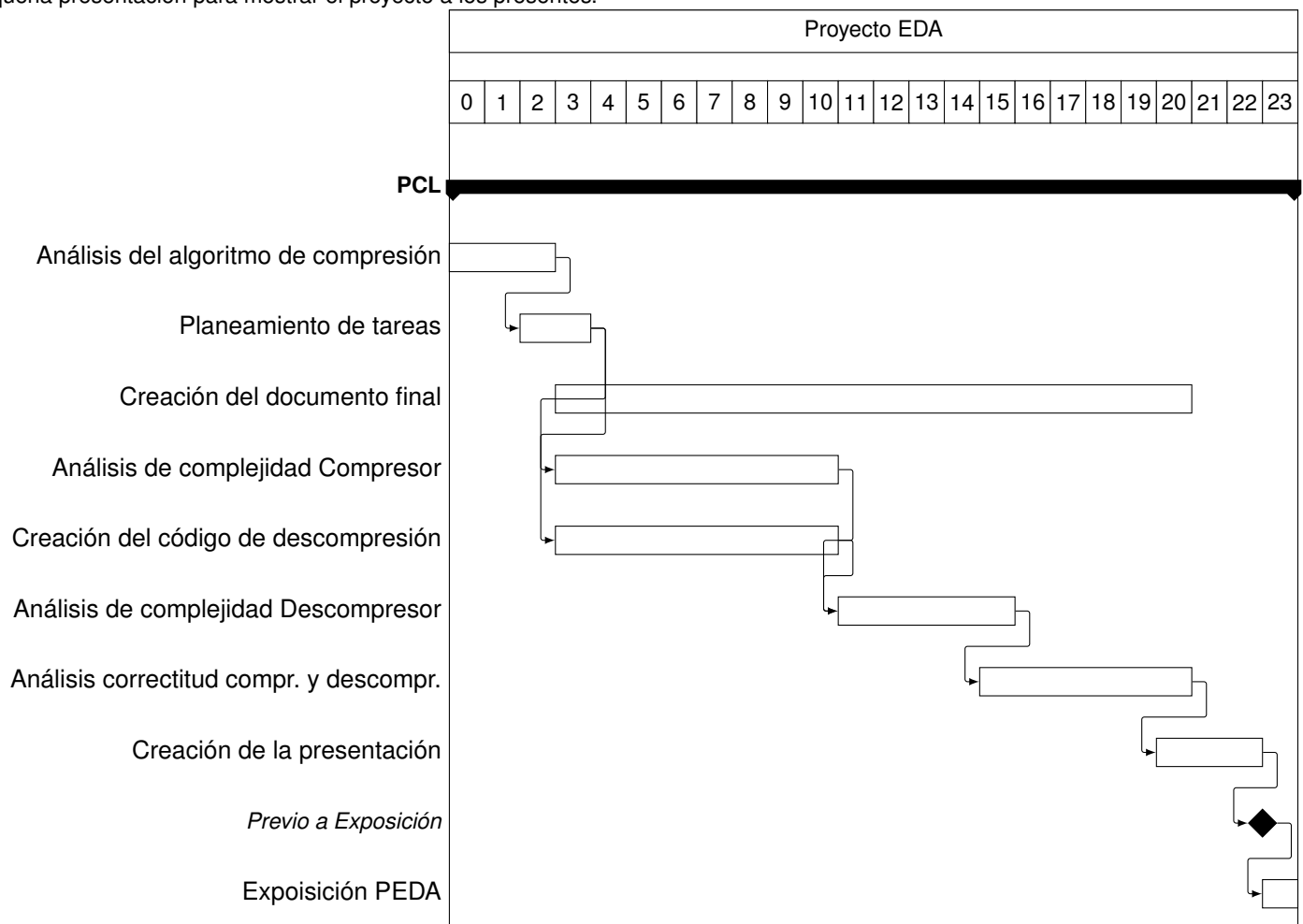
I CICLO 2014



PROPUESTA DE PROYECTO DE ESTRUCTURAS DE DATOS Y ALGORITMOS

ESCUELA DE INGENIERÍA ELÉCTRICA
UNIVERSIDAD DE COSTA RICA

compresión, a la vez que se va a crear el documento, por eso se le va a dar más tiempo para poder crear la tabla en formato \LaTeX para su mejor presentación. A su vez, se comenzará a crear el algoritmo de descompresión que no se había hecho en la presentación de nuestro compañero, para finalizar con su análisis de complejidad. Una vez contruida la tabla y analizado el algoritmo, se trabajará conjuntamente para analizar ambos algoritmos respecto a su correctitud. Finalmente, se van a agregar los diagramas de flujos que no se habían hecho para la presentación de investigación pasada y se procederá a hacer una pequeña presentación para mostrar el proyecto a los presentes.



4. Referencias

1. Press, W., Teukolsky, W., Vetterling, W., Flannery, B. *Numerical Recipes: The Art of Scientific Computing*, 3ra. Ed. Cambridge University Press, 2007
2. Cormen, T., Leiserson, C., Rivest, R., Stein, C. *Introduction To Algorithms*, 3ra. Ed. MIT Press, 2009
3. Sedgewick, R., Wayne, K. *Algorithms*, 4ta. Ed. Pearson Education, 2011
4. Eckel, B. *Thinking in C++, volume I*, 2da. Ed. Prentice Hall, 2000
5. Referencias web:



UNIVERSIDAD DE COSTA RICA
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA ELÉCTRICA
**ESTRUCTURAS ABSTRACTAS DE DATOS Y
ALGORITMOS PARA INGENIERÍA**

IE-0217
I CICLO 2014



PROPUESTA DE PROYECTO DE ESTRUCTURAS DE DATOS Y ALGORITMOS

ESCUELA DE INGENIERÍA ELÉCTRICA
UNIVERSIDAD DE COSTA RICA

- <http://oreilly.com/catalog/masteralgoc/chapter/ch08.pdf>
- www.cs.princeton.edu/rs/AlgsDS07/10Hashing.pdf
- <http://burtleburtle.net/bob/hash/evahash.html>
- <http://cseweb.ucsd.edu/mihir/cse207/w-hash.pdf>
- <http://www.sinfocol.org/herramientas/hashe.php>