# **TÍTULO (DESCRIPCIÓN CORTA DEL PROYECTO. ENTRE 8 Y 12 PALABRAS)**

| Nombre del primer autor  Universidad  País  Dirección e-mail | Nombre del segundo autor  Universidad  País  Dirección e-mail | Mauricio Toro  Universidad Eafit  Colombia  mtorobe@eafit.edu.co |
| --- | --- | --- |

**NOTA DEL DOCENTE: Para ampliar información sobre los requerimientos aquí descritos, consulten la “*Guía para la realización del Proyecto Final de Estructura de Datos 2”* que se entrega. Al final: 1. Borrar este texto escrito en rojo, 2. Adecuar los espacios de los textos, 3. Cambiar el color de los textos a negro. Consideren además que:**

**Textos en negro** = Es todo lo que deben hacer en la entrega 1

**Textos en Verde** = Es todo lo que deben hacer en la Entrega 2

Textos en violeta = Es todo lo que deben hacer en la entrega 3

# **RESUMEN**

Para escribirlo pueden dar respuesta a estas preguntas: ¿Cuál es el problema?, ¿Por qué es importante el problema?, ¿Qué problemas relacionados hay?, ¿Cuál es la solución?, ¿cuáles los resultados? y, ¿Cuáles las conclusiones? Utilizar máximo 200 palabras.

## **Palabras clave**

|  |
| --- |
| Estas son palabras claves que ustedes consideran apropiadas  para indexar el informe PDF en bibliotecas o bases de datos |

## **Palabras clave de la clasificación de la ACM**

|  |
| --- |
| Solo las que están en esta lista de ACM, en <http://bit.ly/2oVE5 2i>  No pueden ser inventadas por ustedes. |

# Ejemplo: Theory of computation → Design and analysis of algorithms → Graph algorithms analysis → Shortest paths

# **1. INTRODUCCIÓN**

Es la justificación de las condiciones en el mundo real que llevan al problema. En otras palabras, es hablar sobre qué va a tratar el documento e incluir la historia de este problema.

# **2. PROBLEMA**

En pocas palabras escriban cuál problema que están resolviendo, además de responder ¿para qué resolver este problema?

## **3. TRABAJOS RELACIONADOS**

## Aquí deberán explicar 4 problemas algorítmicos similares que se encuentren documentados en libros, artículos científicos o sitios web, y dar al menos 1 solución para uno de ellos. NO poner soluciones de tecnología.

## **3.1 Título del primer trabajo relacionado**

Aquí deberán mencionar el primer problema algorítmico relacionado documentado y/o su solución

## **3.2 Título del segundo trabajo relacionado**

Aquí deberán mencionar el segundo problema algorítmico relacionado documentado y/o su solución

## **3.3 Título del tercer trabajo relacionado**

Aquí deberán mencionar el tercer problema algorítmico relacionado documentado y/o su solución

## **3.4 Título del cuarto trabajo relacionado**

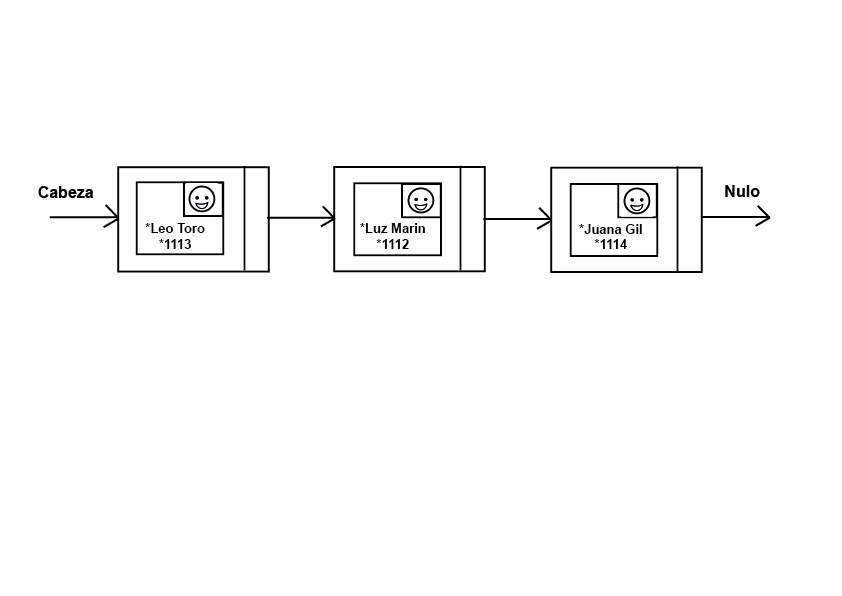
Aquí deberán mencionar el cuarto problema algorítmico relacionados documentado y/o su solución

## **4. TÍTULO DE LA PRIMERA SOLUCIÖN DISEÑADA**

## A continuación, explicamos la estructura de datos y el algoritmo.

## **4.1 Estructura de datos**

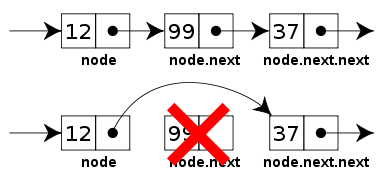
## Diseñen la estructura de datos para resolver el problema y grafíquenla. No usar gráficas extraídas de internet



## **Gráfica 1:** Lista simplemente encadenada de personas. Una persona es una clase que contiene nombre, cédula y foto

## **4.2 Operaciones de la estructura de datos**

## Diseñen las operaciones de la estructura de datos para solucionar el problema eficientemente. Incluyan una imagen explicando cada operación



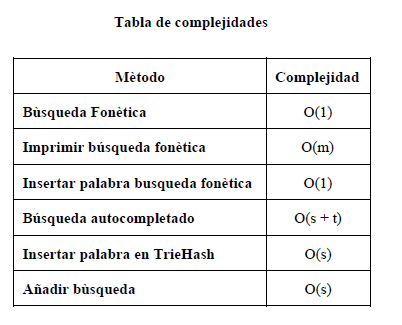
**Gráfica 2:** Imagen de una operación de borrado de una lista encadenada

## **4.3 Criterios de diseño de la estructura de datos**

Expliquen con criterios objetivos, por qué diseñaron así la estructura de datos. Criterios objetivos son, por ejemplo, la eficiencia en tiempo y memoria. Criterios no objetivos y que rebajan la nota son: “me enfermé”, “fue la primera que encontré”, “la hice el último día”, etc. Recuerden: este es el numeral que más vale en la evaulación con 40%

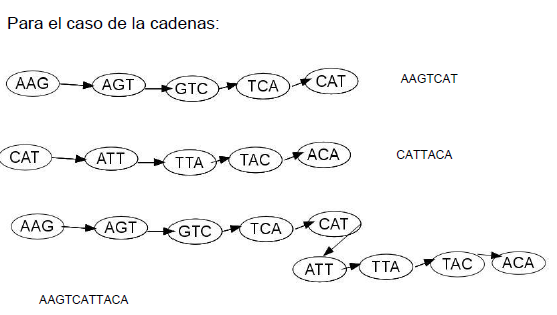
**4.4 Análisis de Complejidad**

Calculen la complejidad de las operaciones de la estructura de datos para el peor de los casos. Vean un ejemplo para reportarla:



**Tabla 1:** Tabla para reportar la complejidad

**4.5 Algoritmo**

Diseñen el algoritmo para resolver el problema y grafíquenlo. No usen gráficas extraídas de internet

**Gráfica 3:** Paso a paso cómo se ensamblan fragmentos de ADN utilizando los grafos de *Bruijn*.

**4.6** **Cálculo de la complejidad del algoritmo**

Calculen la complejidad del algoritmo para el peor de los casos, el mejor de los casos y el caso promedio

|  |  |
| --- | --- |
| **Sub problema** | **Complejidad** |
| Crear el grafo de *Bruijn* con las secuencias | O(S) |
| Actualizar el grafo de *Bruijn* con las secuencias | O(A.V2) |
| Encontrar los genes | O(S) |
| **Complejidad Total** | O(A.S2 + V) |

**Tabla 2:** complejidad de cada uno de los sub problemas que componen el algoritmo. Sea A la longitud de una secuencia de ADN, S el número de secuencias de ADN, y V el número de K-meros diferentes que se obtienen de las secuencias de ADN.

NOTA: Sin complejidad total, el análisis no sirve de nada

**4.7 Criterios de diseño del algoritmo**

Expliquen por qué diseñaron así el algoritmo. Usen criterios objetivos. Criterios objetivos son, por ejemplo, la eficiencia en tiempo y memoria. Criterios no objetivos y que rebajan la nota son: “me enfermé”, “fue la primera que encontré”, “la hice el último día”, etc. Recuerden: este es el numeral que más vale en la evaluación con 40%.

**4.8 Tiempos de Ejecución**

Calculen, (I) el tiempo de ejecución y (II) la memoria usada del algoritmo, para el Conjunto de Datos que está en el ZIP:

## Tomen 100 veces el tiempo de ejecución y memoria de ejecución, para cada conjunto de datos

## 

|  |  |  |  |
| --- | --- | --- | --- |
|  | ***Conjunto de Datos 1*** | ***Conjunto de Datos 2*** | ***...Conjunto de Datos n*** |
| *Mejor caso* | 10 sg | 20 sg | 5 sg |
| *Caso promedio* | 12 sg | 10 sg | 35 sg |
| *Peor caso* | 15 sg | 21 sg | 35 sg |

## **Tabla 3:** Tiempos de ejecución del algoritmo con diferentes conjuntos de datos

## **4.9 Memoria**

Mencionar la memoria que consume el programa para varios ejemplos

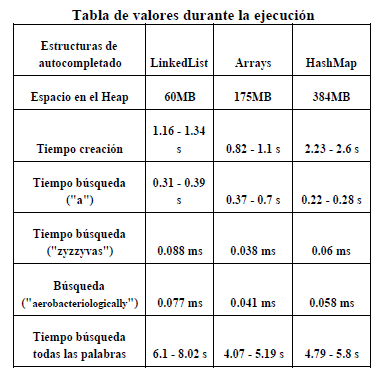
|  |  |  |  |
| --- | --- | --- | --- |
|  | ***Conjunto de Datos 1*** | ***Conjunto de Datos 2*** | ***...Conjunto de Datos n*** |
| **Consumo de memoria** | 10 MB | 20 MB | 5 MB |

## **Tabla 4:** Consumo de memoria del algoritmo con diferentes conjuntos de datos

## Para medir la memoria que consume un programa, se utilizan generadores de perfiles (en Inglés, profilers). Uno muy bueno para Java es VisualVM, desarrollado por Oracle,<http://docs.oracle.com/javase/7/docs/technotes/guides/visualvm/profiler.html> No dejen de usarlo en sus proyectos y en la vida. Para usarlo hay que generar un .jar que es como un ejecutable de Java. En Netbeans "martillo con escoba" y en BlueJ "archivo, generar .jar".

## **4.10 Análisis de los resultados**

Expliquen los resultados obtenidos. Hagan una gráfica con los datos obtenidos, como por ejemplo:



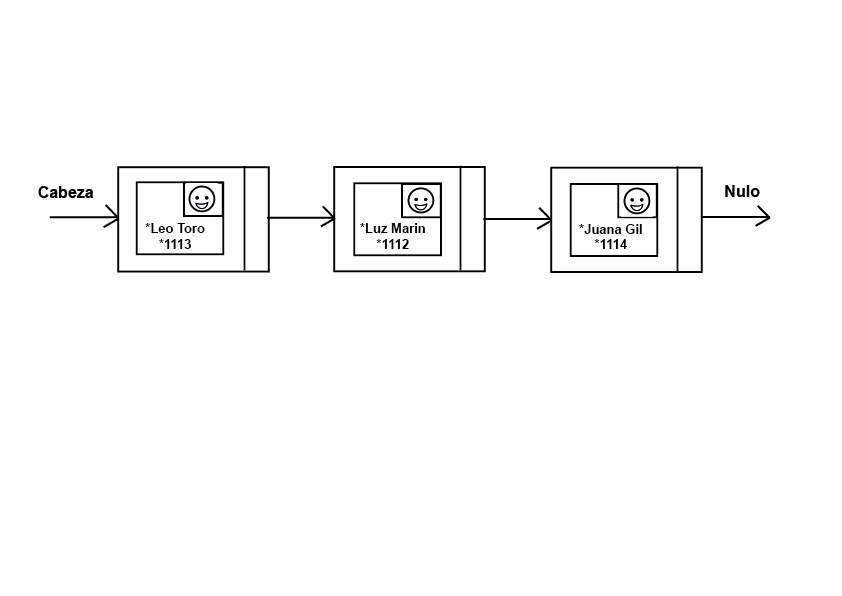
**Tabla 5:** Análisis de los resultados obtenidos con la implementación del algoritmo

**5. TÍTULO DE LA SOLUCIÓN FINAL DISEÑADA**

## A continuación, explicamos la estructura de datos y el algoritmo.

## **5.1 Estructura de datos**

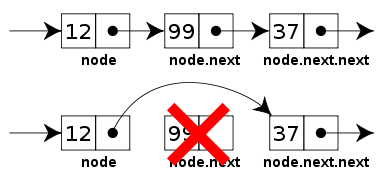
## Diseñen la estructura de datos para resolver el problema y grafíquenla. No usar gráficas extraídas de internet



## **Gráfica 4:** Lista simplemente encadenada de personas. Una persona es una clase que contiene nombre, cédula y foto

## **5.2 Operaciones de la estructura de datos**

## Diseñen las operaciones de la estructura de datos para solucionar el problema eficientemente. Incluyan una imagen explicando cada operación



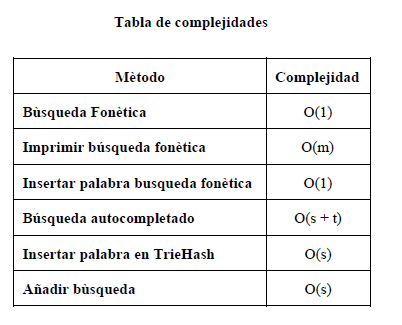
**Gráfica 5:** Imagen de una operación de borrado de una lista encadenada

## **5.3 Criterios de diseño de la estructura de datos**

Expliquen con criterios objetivos, por qué diseñaron así la estructura de datos. Criterios objetivos son, por ejemplo, la eficiencia en tiempo y memoria. Criterios no objetivos y que rebajan la nota son: “me enfermé”, “fue la primera que encontré”, “la hice el último día”, etc. Recuerden: este es el numeral que más vale en la evaluación con 40%

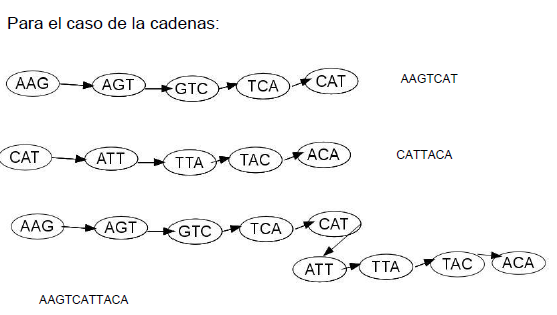
**5.4 Análisis de Complejidad**

Calculen la complejidad de las operaciones de la estructura de datos para el peor de los casos. Vean un ejemplo para reportarla:



**Tabla 6:** Tabla para reportar la complejidad

**5.5 Algoritmo**

Diseñen el algoritmo para resolver el problema y grafíquenlo. No usen gráficas extraídas de internet

**Gráfica 6:** Paso a paso cómo se ensamblan fragmentos de ADN utilizando los grafos de *Bruijn*.

**5.6** **Cálculo de la complejidad del algoritmo**

Calculen la complejidad del algoritmo para el peor de los casos, el mejor de los casos y el caso promedio

|  |  |
| --- | --- |
| **Sub problema** | **Complejidad** |
| Crear el grafo de *Bruijn* con las secuencias | O(N) |
| Actualizar el grafo de *Bruijn* con las secuencias | O(A.N2) |
| Encontrar los genes | O(V) |
| **Complejidad Total** | O(A.N2 + V) |

**Tabla 7:** complejidad de cada uno de los sub problemas que componen el algoritmo. Sea A la longitud de una secuencia de ADN, N el número de secuencias de ADN, y V el número de K-meros diferentes que se obtienen de las secuencias de ADN.

**5.7 Criterios de diseño del algoritmo**

Expliquen por qué diseñaron así el algoritmo. Usen criterios objetivos. Criterios objetivos son, por ejemplo, la eficiencia en tiempo y memoria. Criterios no objetivos y que rebajan la nota son: “me enfermé”, “fue la primera que encontré”, “la hice el último día”, etc. Recuerden: este es el numeral que más vale en la evaluación con 40%

.

**5.8 Tiempos de Ejecución**

Calculen, (I) el tiempo de ejecución y (II) la memoria usada del algoritmo, para el Conjunto de Datos que está en el ZIP:

## Tomen 100 veces el tiempo de ejecución y memoria de ejecución, para cada conjunto de datos

## 

|  |  |  |  |
| --- | --- | --- | --- |
|  | ***Conjunto de Datos 1*** | ***Conjunto de Datos 2*** | ***...Conjunto de Datos n*** |
| *Mejor caso* | 10 sg | 20 sg | 5 sg |
| *Caso promedio* | 12 sg | 10 sg | 35 sg |
| *Peor caso* | 15 sg | 21 sg | 35 sg |

## **Tabla 8:** Tiempos de ejecución del algoritmo con diferentes conjuntos de datos

## Para medir la memoria que consume un programa, se utilizan generadores de perfiles (en Inglés, profilers). Uno muy bueno para Java es VisualVM, desarrollado por Oracle,<http://docs.oracle.com/javase/7/docs/technotes/guides/visualvm/profiler.html> No dejen de usarlo en sus proyectos y en la vida. Para usarlo hay que generar un .jar que es como un ejecutable de Java. En Netbeans "martillo con escoba" y en BlueJ "archivo, generar .jar".

## **5.9 Memoria**

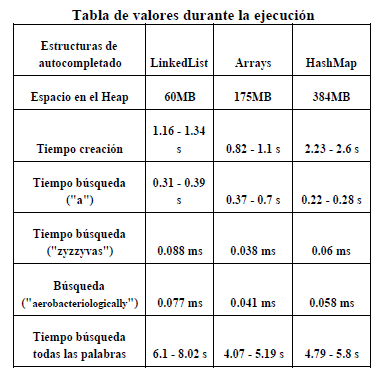
Mencionar la memoria que consume el programa para varios ejemplos

|  |  |  |  |
| --- | --- | --- | --- |
|  | ***Conjunto de Datos 1*** | ***Conjunto de Datos 2*** | ***...Conjunto de Datos n*** |
| **Consumo de memoria** | 10 MB | 20 MB | 5 MB |

## **Tabla 9:** Consumo de memoria del algoritmo con diferentes conjuntos de datos

## **5.10 Análisis de los resultados**

Expliquen los resultados obtenidos. Hagan una gráfica con los datos obtenidos, como por ejemplo:



**Tabla 10:** Análisis de los resultados obtenidos con la implementación del algoritmo

**6. CONCLUSIONES**

Para escribirlas, procedan de la siguiente forma: 1. En un párrafo escriban un resumen de lo más importante que hablaron en el reporte. 2. En otro expliquen los resultados más importantes, por ejemplo, los que se obtuvieron con la solución final. 3.Luego, comparen la primera solución que hicieron con los trabajos relacionados y la solución final. 4. Por último, expliquen los trabajos futuros para una posible continuación de este Proyecto. Aquí también pueden mencionar los problemas que tuvieron durante el desarrollo del proyecto

**6.1 Trabajos futuros**

Respondan ¿Qué les gustaría mejorar en el futuro? ¿Qué les gustaría mejor al algoritmo, estructura de datos, implementación?

# **AGRADECIMIENTOS**

Identifiquen el tipo de agradecimiento que van a escribir: para una persona o para una institución. Luego, escríbanlo de acuerdo al idioma y tengan en cuenta que: 1. El nombre del docente no va porque él es autor. 2. Tampoco sitios de internet ni autores de artículo leídos con quienes no se han contactado. 3. Los nombres que sí van son quienes ayudaron, compañeros del curso o docentes de otros cursos.

Aquí un ejemplo en inglés: This research was supported/partially supported by [Name of Foundation, Grant maker, Donor].

We thank for assistance with [particular technique, methodology] to [Name Surname, position, institution name] for comments that greatly improved the manuscript.

# **REFERENCIAS**

Referenciar las fuentes usando el formato para referencias de la ACM. Léase en <http://bit.ly/2pZnE5g> Vean un ejemplo:

1.Adobe Acrobat Reader 7, Be sure that the references sections text is Ragged Right, Not Justified. <http://www.adobe.com/products/acrobat/>.

2. Fischer, G. and Nakakoji, K. Amplifying designers’ creativity with domainoriented design environments. in Dartnall, T. ed. Artificial Intelligence and Creativity: An Interdisciplinary Approach, Kluwer Academic Publishers, Dordrecht, 1994, 343-364.