Logotipo

Descripción generada automáticamente

UNIVERSIDAD DE LA FRONTERA

FACULTAD DE INGENIERÍA, CIENCIAS Y ADMINISTRACIÓN

DEPARTAMENTO DE INGENIERÍA ELÉCTRICA

**“Estructura de Notas”**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
TAREA 1

ESTRUCTURA DE DATOS Y ALGORITMOS

Profesor: Matthias Clein Espinoza

GRUPO Nº1

SEGUNDO SEMESTRE, AÑO 2021

TAREA 1 – ESTRUCTURA DE DATOS Y ALGORITMOS

**TÍTULO: Estructura de Notas**

NOMBRE: Cristopher Andrés Almonacid Castillo

MATRÍCULA: 20989792K20

CARRERA: ICT

NIVEL CURSADO: 2

NOMBRE: Martin Tomás Canario Dauros

MATRÍCULA: 201577500019

CARRERA: ICT

NIVEL CURSADO: 3

MATTHIAS IGNACIO CLEIN ESPINOZA

Profesor de la Asignatura

RAMIRO ADRIAN DONOSO FLOODY

Director de Carrera

PABLO FRANCISCO MEZA NARVÁEZ

Director de Departamento

**Índice de contenido**

[**Introducción** 4](#_Toc83604060)

[**Desarrollo** 4](#_Toc83604061)

[**Conclusiones** 5](#_Toc83604062)

[**Nomenclatura** 5](#_Toc83604063)

[**Bibliografía** 5](#_Toc83604064)

# **Introducción**

Comenzamos este trabajo con la ilusión de crear un programa que logre categorizar notas de texto. Esto serviría enormemente a las distintas usuarias y usuarios potenciales de algún software de notas, en el que necesiten ordenar estas las cuales estarán almacenadas en una base de datos. Ya que, hoy en día, se desconoce un programa que sea tan liviano, preciso y eficiente, para lo requerido. Por suerte, pronto será mañana, y veremos cómo Estructura de Notas, cambia la vida de las personas.

Una vez terminado el análisis sobre entradas, salidas, y problemas; iniciamos el desarrollo del código creando una interfaz para darle al usuario las distintas opciones, como: listar las notas alfabéticamente, por fecha, eliminar alguna, o listar las categorías. Posteriormente, se realizó una estructura de datos para poder categorizarlas según las etiquetas desarrolladas, y así, finalmente crear un algoritmo que permita desplegar un menú para conocer las notas, el manejo de archivos para abrir el correspondiente, y listarlas de forma sistematizada.

# **Desarrollo**

El objetivo de nuestro trabajo es lograr la realización de un código que agrupe notas según distintas etiquetas que estas lleven, para así, tener un acceso rápido a estas en el momento en que las necesitemos acceder a ellas posteriormente.

Para ello desarrollamos 5 archivos, los cuales ayudan a que el código principal funcione. El anteriormente mencionado, se llama *main.cpp*, como en la mayoría de los casos, el cual requiere del *makefile* para automatizar la compilación de los archivos. *ednStruct.h* da la estructura del programa, mientras que el *edn.h*, agrega las librerías. e*dn.cpp* habla de las implementaciones, y finalizamos con el readme.md, quien incluye toda la documentación.

Hemos de investigar sobre el “lenguaje de marcado”, el cual será el lenguaje sobre el cual escribimos las notas por su amplia capacidad de desarrollo sin escapar del editor de código (Visual Studio Code).

El código está escrito en C++, ya que a pesar de que estudiamos C/C++ durante 8 clases, para escanear cadenas de texto, no sólo tuvimos que recurrir a métodos de binario, sino que también pasamos completamente a C++, debido a que no encontramos solución a los cortes de lectura que obteníamos con C al momento en el que ingresaba una tecla espacio a través de la nota analizada.

# **Conclusiones**

* Los archivos utilizados en el proyecto han sido de gran ayuda en algunos casos, y sumamente necesarios en otros. Mientras más crece un código, mayores son sus requerimientos, y la correcta utilización de nuestras herramientas en conjunto con “las buenas prácticas”, permitieron que acabemos de modo más eficaz y eficiente el código.
* El lenguaje de marcado, es un lenguaje de sintaxis de documentos de texto, el cual nos permite elaborar documentos de texto, agregando títulos, subtítulos, párrafos, etcétera, sin necesidad de recurrir a otro programa editor de texto.
* El mundo de la programación es inmenso, tan solo intenta vislumbrar que existen más de 675 lenguajes de programación. Es por esto, que ante limitaciones como las que sufrimos durante el desarrollo del software, retomar las clases grabadas, retomar los libros anclados en nuestro campus virtual, y al acompañar esto de extensas búsquedas en foros y reciclaje de código, podemos encontrarnos siempre con estupendas fuentes de información. El desarrollo de tu software nunca estará perdido, posiblemente si estancado, pero mantente atenta o atento, que adversidades como un error de compilación, o la imposibilidad de ejecutar unas líneas de código, no detendrán la máquina creativa que se genera entre tú y el teclado.

En conjunto con lo anteriormente expuesto, y las ya mencionadas clases de EDD, podemos concluir que se puede desarrollar un software que permite al usuario de forma sencilla, tener una lista organizada de notas para su posterior referencia. Obteniendo así un filtro eficiente de tiempo para tus necesidades.

Combinando estos dos elementos, nos permitimos crear un ecosistema óptimo para el desarrollo de un conjunto de notas ordenadas, categorizadas y sencillas de escribir.

# **Nomenclatura**

*.cpp* – Formato de archivo para C++ (proveniente de c plus plus)-

*.h* – Formato de archivo para headers.

# **Bibliografía**

Aguinaga, I. Aybar, M. Bustamante, P. Olaizola, L. Lacazan, I. (2004) "Aprenda C++ Avanzado como si estuviera en primero". Campus tecnológico de la Universidad de Navarra, escuela superior de ingenieros.

Aguinaga, I. Aybar, M. Bustamante, P. Olaizola, L. Lacazan, I. (2004) "Aprenda C++ Básico como si estuviera en primero". Campus tecnológico de la Universidad de Navarra, escuela superior de ingenieros.

Fermí, V. "Programación en C/C++ (Manual FV)"

FlavioCopes. How to return a string from a C function. https://flaviocopes.com/c-return-string/. Temuco. Visitada el 25 de septiembre de 2021.

StackOverFlow. Find word in a text in C++ and print some next specific lines. https://stackoverflow.com/questions/33022697/find-word-in-a-text-in-c-and-print-some-next-specific-lines#33023934. Temuco. Visitada el 25 de septiembre de 2021.