|  |  |  |
| --- | --- | --- |
|  | **Carátula para entrega de prácticas** | |
| Facultad de Ingeniería | | Laboratorio de docencia |

Laboratorios de computación

salas A y B

|  |  |
| --- | --- |
| *Profesor:* | Marco Antonio Martinez Quintana |
| *Asignatura:* | Estructura de Datos y Algoritmos I |
| *Grupo:* | 17 |
| *No de Práctica(s):* | 1 |
| *Integrante(s):* | Enzo Valdés Zavala |
| *No. de Equipo de cómputo empleado:* |  |
| *No. de Lista o Brigada:* |  |
| *Semestre:* | 2020-1 |
| *Fecha de entrega:* | 9 de febrero del 2020 |
| *Observaciones:* |  |
|  |  |

CALIFICACIÓN: \_\_\_\_\_\_\_\_\_\_

**Objetivos:**

Comprender el concepto de arreglos dentro de la programación, ya sean unidimensionales o multidimensionales definiendo su tamaño, para poder aplicarlo a problemas de ingeniería computacional.

**Introducción:**

Dentro de los lenguajes computacionales, encontramos el uso de variables para expresar datos dentro del código.  
Uno de estos datos son los arreglos, que son variables capaces de albergar más de un dato al mismo tiempo.  
Estos pueden ser unidimensionales, o multidimensionales.

Para comprender como funcionan, tendríamos que ubicar un arreglo como un espacio geométrico, como un cuadrado, compuesto de solo columnas, filas o más categorías dependiendo de las dimensiones de este.  
Cada casilla, representada por un espacio de la fila/columna, da lugar a un dato en particular, teniendo este su propio espacio y “coordenada”.  
Para acceder a este dato, tendremos que entender la dimensión del arreglo que empieza en 0 y termina en n-1, donde n es el tamaño de la dimensión especificada por el programador.

Dentro de los arreglos, existen los continuos y los ligados.  
Los arreglos continuos son aquellos que se crean desde el inicio del programa y son estáticos, es decir, no se redimensionan de ninguna forma.

Los arreglos ligados son aquellos que se declaran en el tiempo de ejecución, por lo que se pueden redimensionar durante su ejecución para usar de manera mas eficiente la memoria (que este tipo de memoria es conocida como memoria dinámica).

**Desarrollo:**

**Explicación:**

El siguiente código esta basado en el principio de encriptación y de cifrado de un antiguo método de criptografía espartano, llamada la escítala espartana.

Este consiste en enrrollar una tira de escritura a lo largo de un palo (llamado escítala) y escribir sobre la tira una vez enrrollada.  
Al desenrollar el mensaje, no tiene sentido a menos que se posea una escitala similar a la usada para crear el mensaje.

Asi que nuestro código se dividirá en 2 partes, la función main que incluirá un mini menú que por medio de un switch seleccionara la segunda parte del código, la encriptación o descifrado de un mensaje en caracteres.

La segunda parte del código se basa en crear dos funciones void (aquellas que no regresan ningún valor), llamadas “**crearMensaje**” y “**descifrarMensaje**”.

La función “crearMensaje” busca crear un mensaje por medio de un arreglo de columnas y renglones, donde será de tipo char llamado **escitala**.

Este arreglo es de dos dimensiones, mientras el otro que vamos a crear **texto** es solo de una dimensión.

Luego, pediremos al usuario que escriba el mensaje y lo almacenaremos en el arreglo “texto”.

A partir de aquí, lo vamos a cifrar por medio de dos ciclos for, donde recorrerá el arreglo “escitala” para ajustar el tamaño del arreglo “texto”.

De ahí, lo ciframos con otros dos ciclos for, donde mezclamos renglones con columnas y viceversa.  
Asi, los datos que esten guardados en las columnas, aparecerán el los espacios de los renglones y viceversa.

La otra función “descifrarMensaje” empezara igual que la función anterior, definiendo los arreglos “escitala” y “texto”.

Pero esta vez, descifrara el mensaje cifrado por medio de otros 4 ciclos for, donde los 2 primeros recorren la escitala y ajustan el texto, y los últimos 2 acomodaran los datos mezclados que cifro la función anterior.

Asi, podemos emular una escitula espartana.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A close up of a logo

Description automatically generated

**Conclusión:**

Se cumplio el objetivo de la practica, ya que entendimos el concepto de los arreglos, su profundidad con las diferentes dimensiones, su forma de interactuar con la memoria (siendo estos continuos o ligados) y terminamos aplicandolos para emular una escitala espartana.

**Bibliografía:**

-El lenguaje de programación C. Brian W. Kernighan, Dennis M. Ritchie, segunda edición, USA, Pearson Educación 1991.

-Teoria vista en el salón de clases.