

Proyecto IV

- 👉 Para realizar este proyecto debes haber estudiado previamente todo el material a investigar de la Unidad más Árboles y Grafos.
- 👉 Una vez terminado el proyecto, comprime la carpeta que contiene el desarrollo de los requerimientos solicitados (implementación de métodos e implementación de Juego, block de notas con los integrantes del grupo) comprimir la carpeta y sube el .zip
- 👉 Fecha de entrega Miércoles 20/11/2024.
- 👉 Exposición: 25/11/2024 (Deben exponer todos los integrantes del grupo, apoyándose con diapositivas o pizarra explicando cada tema.)
- 👉 Desarrollo proyecto:
 - El proyecto se debe desarrollar de manera grupal.
 - Se evaluará avance del proyecto, debe comprimir la carpeta y subirla en las clases que corresponda. (10 puntos)

Unidad IV

- 👉 El proyecto está basado en los siguientes temas:
 - Complejidad de Algoritmos.
 - Algoritmo de búsqueda y Ordenamiento.

I. Realizar un Informe en no más de 8 planas en donde exponga los temas a continuación:(30 puntos)

1. Complejidad de Algoritmo
 - a. Que es la complejidad de algoritmo
 - b. Eficiencia de un algoritmo y aplicaciones
 - c. Notación O-Grande y sus propiedades
 - d. Notación Asintótica
 - e. Tiempo de ejecución de un algoritmo.
2. Métodos de búsqueda
 - a. Por selección
 - b. Método de la Burbuja
 - c. Método Radix
 - d. Método Quicksort (rápido),
 - e. Ordenación por Inserción
 - f. Método ShellSort
 - g. Ordenamiento por Mezcla
3. Implementación
 - a. La implementación de cada método de búsqueda en C++.
 - b. Realizar la comparación de los métodos de búsqueda con la misma entrada de datos aproximadamente con 10.000 o más datos (estos datos deben ser al azar y desordenados), para visualizar bien cada método y tiempo de ejecución.
 - c. Realizar análisis e indicar la eficiencia de cada método utilizando con (Big-O)

II. Crear un juego de búsqueda de parejas. (30 puntos)

Descripción

Este juego se basará en la búsqueda alfabética de letra asociado a una palabra que comience con dicha letra. Ejemplo A – Ana, B- Bote

Para este juego necesitará una matriz en donde de tamaño 6(filas) x 6(columnas), tipo string. En ella almacenara internamente los valores de la forma: (solo referencia)

A	Q	B	P	C	OSO
ZORRO	ARBOL	PALA	BOLA	Y	CASA
E	QUEQUE	F	R	G	ROCA
YEMA	ELEFANTE	WIFI	FOCA	O	GATO
I	W	J	U	K	MAMA
Z	IGLU	VACA	JARRA	TATA	KOALA
LL	UVA	LLAMA	V	M	T

En pantalla solo se vera la matriz con su posición, color o separación de cada celda

1	2	3	4	5	6
7	8	9	10	11	12
13	14	15	16	17	18
19	20	21	22	23	24
25	26	27	28	29	30
31	32	33	34	35	36

De manera si indica el jugador posición 1 y posición 8 debería aparecer en pantalla si corresponden los valores, en este ejemplo corresponden deben quedar en pantallas.

A	2	3	4	5	6
7	ARBOL	9	10	11	12
13	14	15	16	17	18
19	20	21	22	23	24
25	26	27	28	29	30
31	32	33	34	35	36

Si no corresponden solo debe mostrarse el contenido de las ubicaciones indicadas por 5 segundos y luego volver a dejarlas como estaban.

Como resultado indicar el tiempo que demora en terminar el juego.

Luego de acertar a todas las parejas, indicar en un mensaje que se ordenara la matriz y debe mostrar la matriz ordenada por: LETRA – PALABRA

Ejemplo:

A	ARBOL	B	BOLA	C	CASA
E	ELEFANTE	...			
...					

Para ordenar debe elegir entre los siguientes métodos de ordenamiento

- Por selección*
- Método Radix*
- Método Quicksort (rápido),*
- Ordenación por Inserción*
- Método Shellsort*

Comente el código en donde realizo el ordenamiento y que tipo de ordenamiento utilizo.

***** **Mostrar por pantalla cada salida del juego.** *****

III– Crear el Juego de Laberinto. (30 puntos)

Descripción

Este juego será para 2 jugadores, en donde el jugador que recorra el camino más corto ganará la Corona. Se tiene un laberinto de $n = 30$ y $m = 30$, la posición inicial será la (0,0), desde ahí deberá recorrer el laberinto de manera vertical u horizontal.

En donde encontrara con obstáculos (■ o X), 10 tesoros (T) y el premio mayor una corona de oro (C).

El laberinto tendrá varios caminos opcionales para llegar a la salida en donde lo espera la corona.

Para obtener tesoros, deberá llegar a él y se contará el tesoro al jugador, luego de obtenerlo se marcará con una ■ o X de obstáculo.

Continua por el laberinto hasta llegar a la corona donde deberá contestar una pregunta cuya respuesta debe ser Y/N y luego indicar si gana o no la corona, más el detalle de todos los tesoros que gano. Junto con ello indicar el valor del camino que tomo.

Luego se inicia el juego para el siguiente jugador, se cargará en pantalla el laberinto y del mismo modo evaluar al jugador dos, y al terminar indicar si gana o no la corona, más el detalle de todos los tesoros que gano. Junto con ello indicar el valor del camino que tomo.

Al final mostrar una tabla comparativa con los tesoros obtenidos, corona si la obtuvo y el valor de camino más corto.

Gana el jugador que haya obtenido el camino más corto.

Para calcular el camino más corto considere cada movimiento el valor de 5 horizontal y 10 Vertical

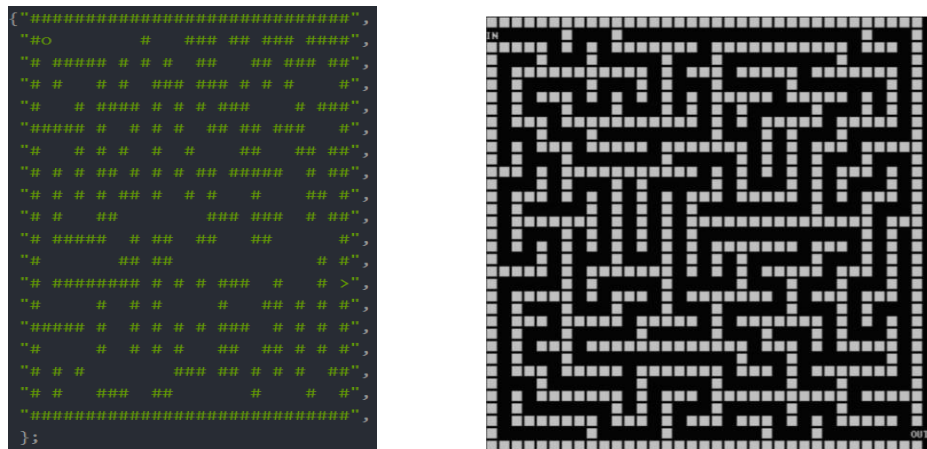
Recuerde tener almacenadas las preguntas para la corona y mostrar al azar la pregunta para que el jugador que llegue a ella conteste. Si contesta correctamente se sumará 7 puntos al total de lo contrario 0. Cada tesoro tendrá un puntaje de 3 puntos.

Almacenar al menos 3 mapas de laberintos con distintos caminos para que el usuario quiera decidir jugar otro juego y vaya variando entre los mapas si continúan jugando.

Los laberintos debes ser cargados al azar, no los elije el jugador.

Foto ejemplo de mapa de un laberinto. (Referencial)

Fig. 1



******* Se evaluará vocabulario adecuado, faltas ortográficas y en los mensajes visualizados en pantalla en el Juego. Con 10 puntos menos*******