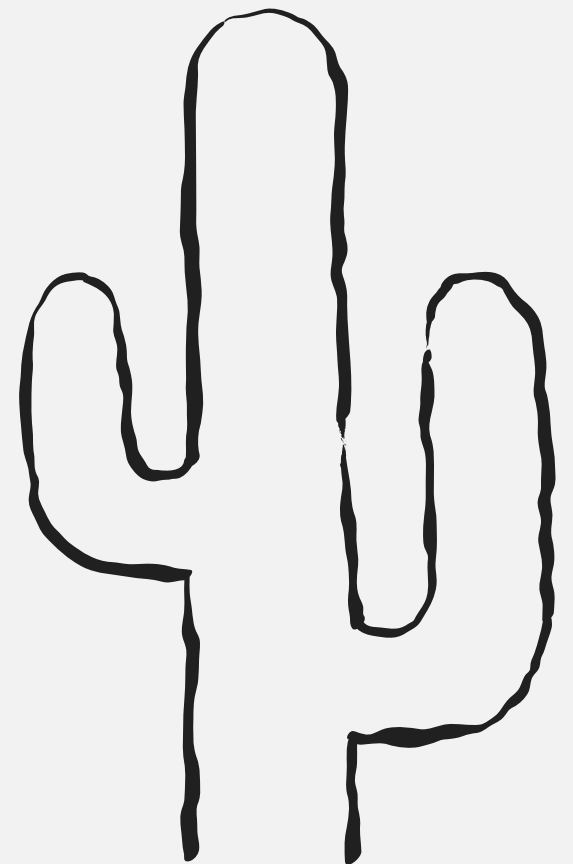
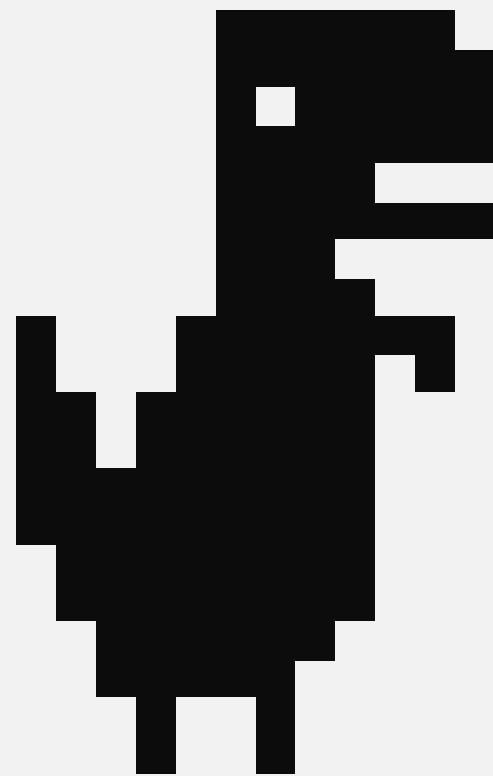
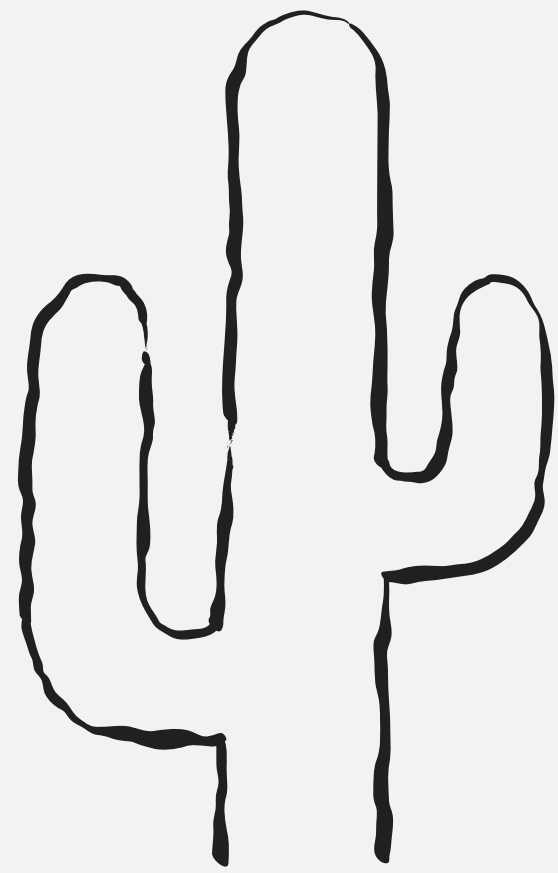


# **DINO SURVIVAL**

***HECHO POR***

***Moises Sebastian Marin Erazo***

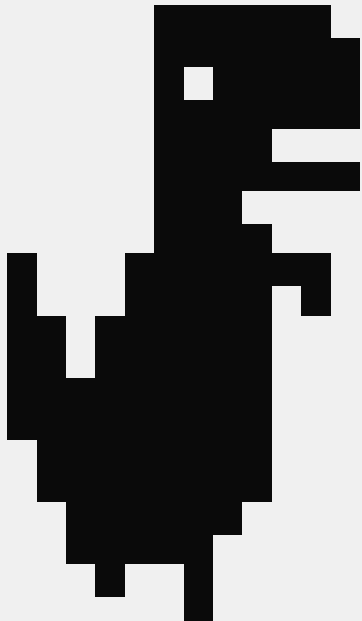
***April Oriana Corpus Coba***



# CONTENIDO

- 1 Introducción
- 2 Descripción
- 3 Bibliotecas utilizadas
- 4 Clases utilizadas
- 5 Complicaciones
- 6 Bibliografia

```
      *****
    *****  **      *****  **
  ***          *  ***          *
*****          *****
```



```
| ** |
[****]
| ** |
| ** |
```



Sin Internet

Intenta:

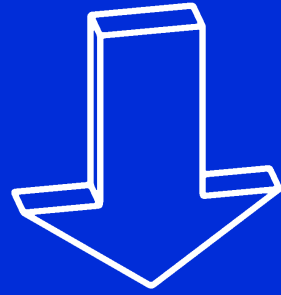
- Comprobar los cables de red, el módem y el router
- Volver a conectarte a Wi-Fi

## INTRODUCCIÓN

El juego Dino Survival nació de la idea de imitar el reconocido juego “Chrome Dino”, que aparece en el navegador Google Chrome cuando el usuario no posee conexión a Internet.

Dino Survival fue programado en C++, utilizando diversas bibliotecas que se explicarán más adelante.

# DESCRIPCIÓN



BIENVENIDO AL JUEGO DINO SURVIVAL

CREADO POR APRIL Y MOISES

ASEGURATE DE MAXIMIZAR LA VENTANA

Y PRESIONA ENTER PARA EMPEZAR



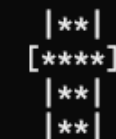
Al comenzar, se muestra una pantalla de bienvenida donde se dan crédito a los autores. Para comenzar a jugar, debes presionar ENTER.

La dinámica es sencilla: debes presionar la flecha hacia arriba para que el dinosaurio salte sobre los cactus que irán apareciendo, y con el paso del tiempo irás sumando puntos. En caso de colisionar pierdes y deberás comenzar de nuevo.

# DESCRIPCIÓN

El juego cambia entre el día y la noche cada 700 puntos, mientras que cada 400 los cactus sufrirán un cambio de velocidad aleatorio que probarán tus reflejos.

A partir de los 2000 puntos, cada 400 también se mostrará un cielo de color rojo, que representa el peligro de extinción que corre nuestro dinosaurio, pero seguirá alternando entre día y noche.



## **IOSTREAM**

Para mostrar texto y caracteres especiales ASCII.

## **FSTREAM**

Manejo de archivos. Utilizada para guardar y leer el puntaje máximo del jugador.

## **THREAD**

Provee hilos o “threads” para ejecución simultánea de funciones. Es decir, ejecutar varias funciones a la vez.

## **CSTDLIB**

Para utilizar la función rand y srand, que generan numeros aleatorios.

# **BIBLIOTECAS UTILIZADAS**

## **MUTEX**

Evita conflictos cuando dos hilos intentan usar la misma función a la vez.

## **WINDOWS.H**

Funcionalidades del sistema operativo Windows. Permite controlar las características de la consola y de la ventana, así como comprobar si una tecla está presionada.

## **LIST**

Para la creación de una lista que almacena las nubes que se mueven en el cielo.

## **CTIME**

Obtener el tiempo. En realidad, se usa para inicializar la semilla aleatoria y generar numeros aleatorios.

# CLASES UTILIZADAS

Dinosaurio
<ul style="list-style-type: none"><li>- x: int</li><li>- y: int</li></ul>
<ul style="list-style-type: none"><li>+ Constructor(int x, int y)</li><li>+ pintar(): void</li><li>+ borrar(): void</li><li>+ moverPata1(): void</li><li>+ moverPata2(): void</li><li>+ saltar(int cuanto): void</li><li>+ X(): int</li><li>+ Y(): int</li></ul>

utilizado por

Cactus
<ul style="list-style-type: none"><li>- x, y: int</li><li>- anchoConsola: int</li><li>- tipo: int</li><li>- altura, ancho: int</li></ul>
<ul style="list-style-type: none"><li>+ Constructor(int x, int y)</li><li>+ pintar(): void</li><li>+ borrar(): void</li><li>+ mover(): void</li><li>+ colision(Dinosaurio): bool</li><li>+ establecerDimensiones(): void</li><li>+ X(): int</li><li>+ Y(): int</li><li>+ vel: int static</li></ul>

Puntaje
<ul style="list-style-type: none"><li>- puntaje: int</li><li>- colorBlanco: bool</li></ul>
<ul style="list-style-type: none"><li>+ Constructor()</li><li>+ incrementar(): void</li><li>+ mostrar(): void</li><li>+ verificarColor(): void</li><li>+ obtenerPuntaje(): int</li></ul>

Nube
<ul style="list-style-type: none"><li>- x, y: int</li><li>- anchoConsola: int</li></ul>
<ul style="list-style-type: none"><li>+ Constructor(int x, int y)</li><li>+ pintar(): void</li><li>+ borrar(): void</li><li>+ mover(): void</li></ul>

# **COMPLICACIONES**

**Cambio de la  
idea inicial de  
juego debido a  
complejidad**

**Problema  
con ejecucion  
simultanea  
de funciones**

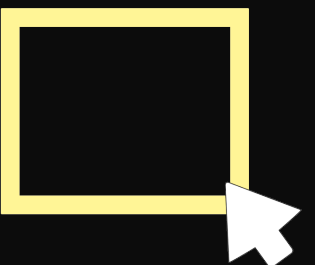
**Problema  
con el uso  
compartido  
de printf  
entre hilos**

**Optimizacion  
de pantalla  
debido a las  
dimensiones  
del juego**



# BIBLIOGRAFÍA

- La biblioteca iostream — Fundamentos de Programación en C++. (s. f.).  
[https://www2.eii.uva.es/fund\\_inf/cpp/temas/3\\_entrada\\_salida/iostream.html](https://www2.eii.uva.es/fund_inf/cpp/temas/3_entrada_salida/iostream.html)
- Nivardo. (2024, 11 abril). Archivos en C++. Oregoom.com. <https://oregoom.com/cpp/archivos/>
- C++ – Threads (Hilos). (s. f.). <https://code0x66.blogspot.com/2017/01/c-threads-hilos.html>
- Turbo Codigo (2021, 15 de noviembre). Numeros aleatorios en C++.  
<https://youtu.be/tpqxsFY89uU>
- Microsoft (2023, 16 de junio). Clase list. <https://learn.microsoft.com/es-es/cpp/standard-library/list-class?view=msvc-170>
- mmeverdies (2016, 20 de mayo). Respuesta al foro "¿Cómo usar la función kbhit() repetidas veces?". Stack Overflow. <https://es.stackoverflow.com/a/10512>





**DISFRUTEN EL JUEGO**