

Universidad Autónoma de
Baja California

Programación Estructurada

LOTERIA EN C

1289251

Jesús Villener Arriaga Portillo

Proyecto

27 de Noviembre 2024

01. **CAPÍTULO 1**
Análisis y diseño

02. **CAPÍTULO 2**
Implementación

03. **CAPÍTULO 3**
Ejecución

04. **CONCLUSIONES Y
REFERENCIAS**
Bibliografía y referencias APA 7

TABLA DE CONTENIDO

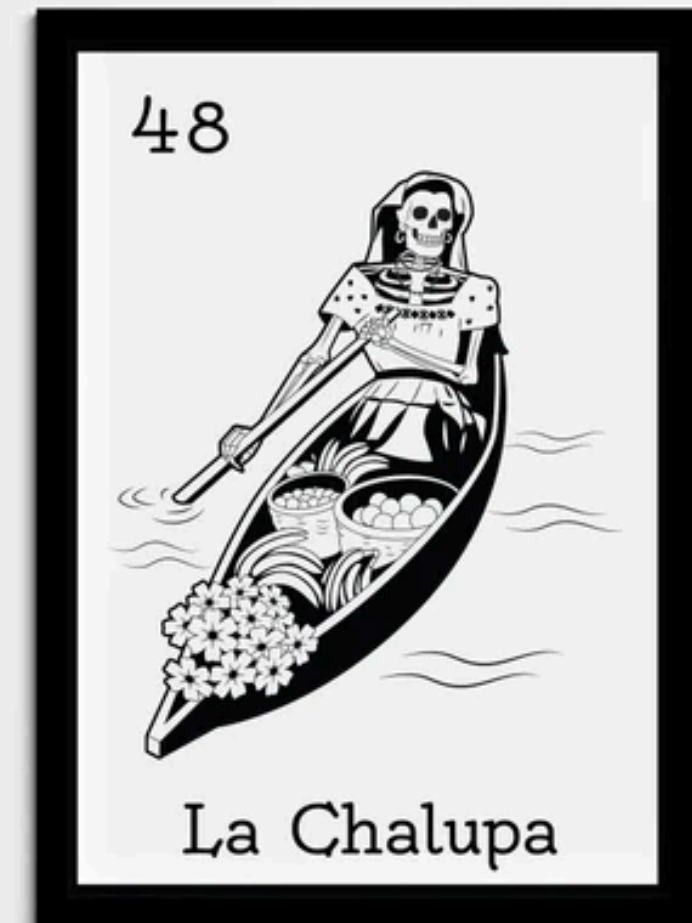
CAPÍTULO 1

ANÁLISIS Y DISEÑO

Definición del juego

La lotería mexicana es un juego de azar nacido en México que consta de un mazo de 54 o 56 cartas y un número indefinido de tarjetas, por lo regular se participa entre 2 y 30 personas. Se utilizan tarjetas con 16 cartas aleatorias, la tabla es de 4x4.

En cada turno se extrae una carta al azar de una baraja, esta se anuncia y solo los participantes que la tengan en su tabla la marcan. El ganador será el primero que llene su tabla.

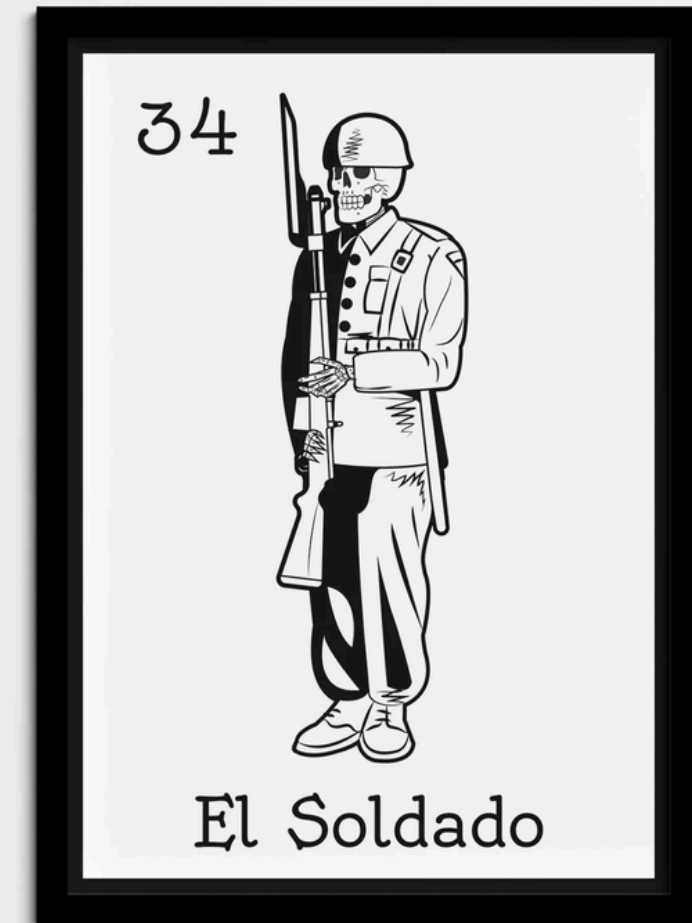


CAPÍTULO 1

ANÁLISIS Y DISEÑO

Reglas Básicas

1. Gana el jugador que llene toda su tabla
2. Una vez que hay un ganador se termina el juego
3. La carta será al azar.
4. Solo se saca una carta al azar por turno

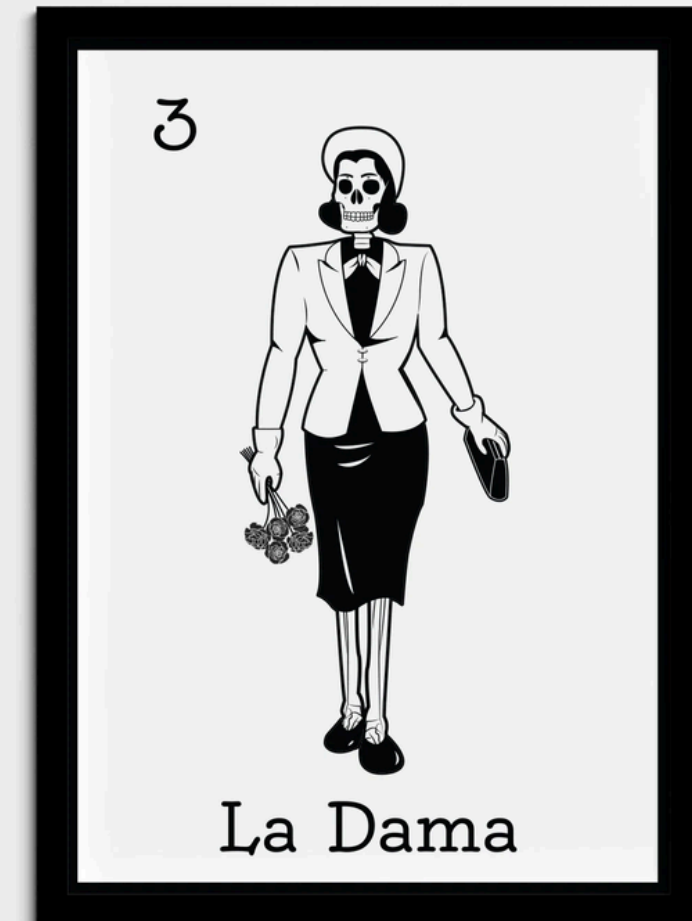


CAPÍTULO 1

ANÁLISIS Y DISEÑO

Funciones principales requeridas

1. Mostrar estado de las tablas
2. Tirar carta
3. Verificar un ganador
4. Salir



CAPÍTULO 2

IMPLEMENTACIÓN

Tipos de datos

Arreglos tipo entero para cada jugador iniciando en 0.
Para la creación de su tabla de lotería.

```
int jugador1[4][4] = {0};  
int jugador2[4][4] = {0};
```

Estructuras de control selectica

Se utilizan estructuras de control selectiva para verificar espacios en las tablas.

```
if(!Verificar_espacio(jugador1))  
    return;
```



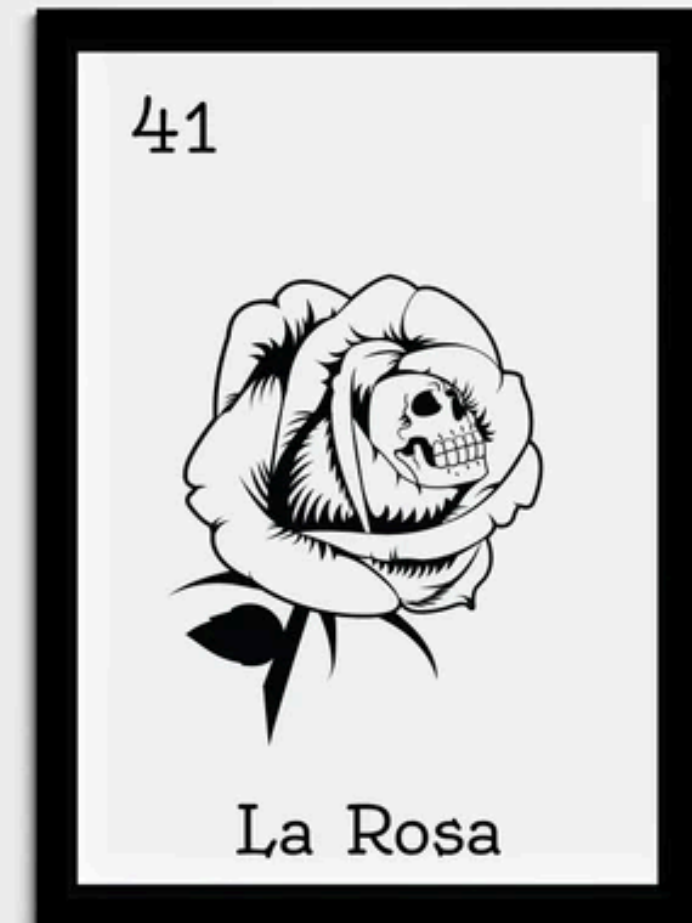
CAPÍTULO 2

IMPLEMENTACIÓN

Estructura de control iterativa

Se utilizan ciclos para llenar la tabla de manera aleatoria haciendo más acorde a la realidad.

```
do{  
    row= rand() % 5;  
    column = rand() % 5;  
}while(jugador1[row][column] != 0);
```



CAPÍTULO 2

IMPLEMENTACIÓN

Funciones modulares

Se utilizan varias funciones fuera del main para llevar mejor el control y creación del programa.

```
//funciones.h
#ifndef FUNCIONES_H
#define FUNCIONES_H
void menu();
void imprimir_matriz(int (*jugador1)[4], int (*jugador2)[4]);
int Aleatorio();
int Verificar_espacio(int(*jugador)[4]);
void Llenar1(int(*jugador1)[4]);
void Llenar2(int(*jugador2)[4]);
#endif
```



CAPÍTULO 3

PRUEBAS

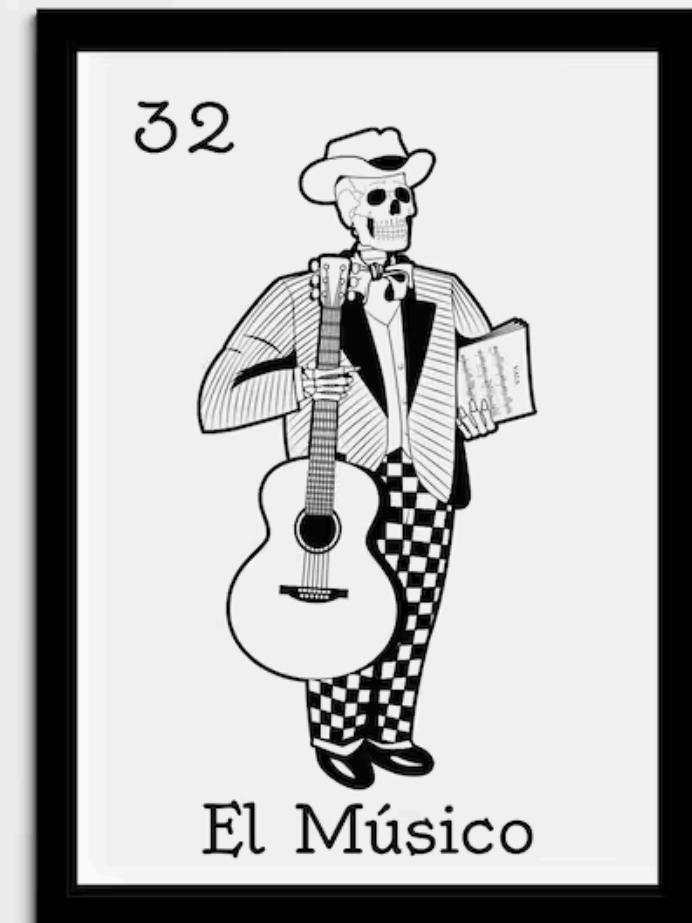
Ejecución

Al iniciar se iniciará con un menú para seleccionar una opción.

```
1.Ver tableros de loteria de jugadores
2.Tirar carta
3.Salir
Opción: █
```

Si introducimos cualquier otro número insistirá en que la opción es incorrecta.

```
1.Ver tableros de loteria de jugadores
2.Tirar carta
3.Salir
Opción:4
Opción no Válida
```



CAPÍTULO 3

PRUEBAS

Ejecución

Al seleccionar la opción 1 mostrara el tablero de los jugadores.

```
1.Ver tableros de loteria de jugadores
2.Tirar carta
3.Salir
Opción:1
Jugador 1:
|0||0||0||0|
|0||0||0||0|
|0||0||0||0|
|0||0||0||0|

Jugador 2:
|0||0||0||0|
|0||0||0||0|
|0||0||0||0|
|0||0||0||0|
```



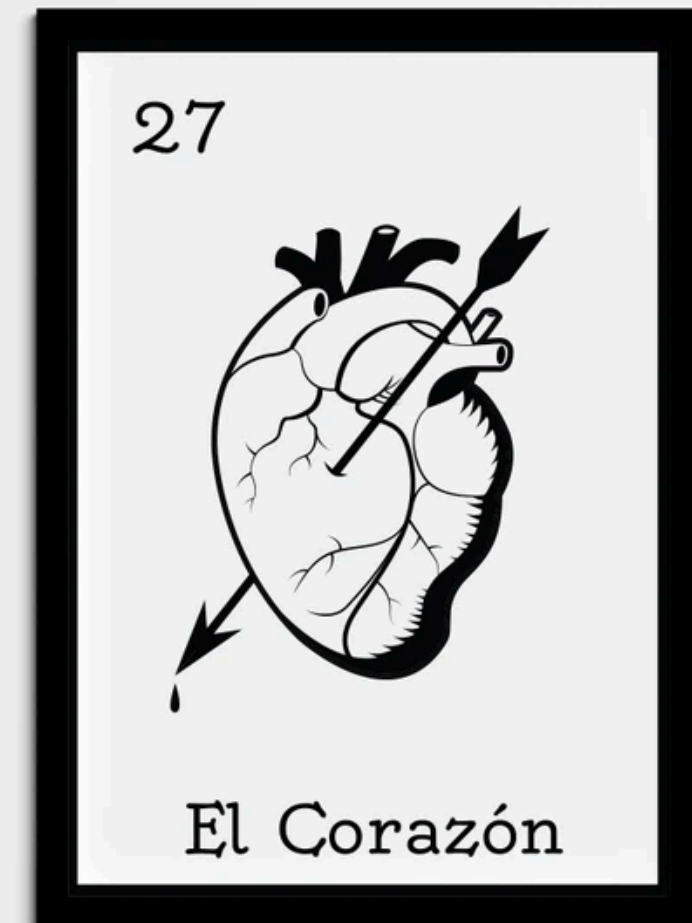
CAPÍTULO 3

PRUEBAS

Ejecución

Al seleccionar la opción 2 se generara un número aleatorio, si el número es par se le asigna al jugador 1 de no ser así al jugador 2.

```
1.Ver tableros de loteria de jugadores
2.Tirar carta
3.Salir
Opción:2
Número generado 10
El número es par. Jugador 1 Marca casilla
```



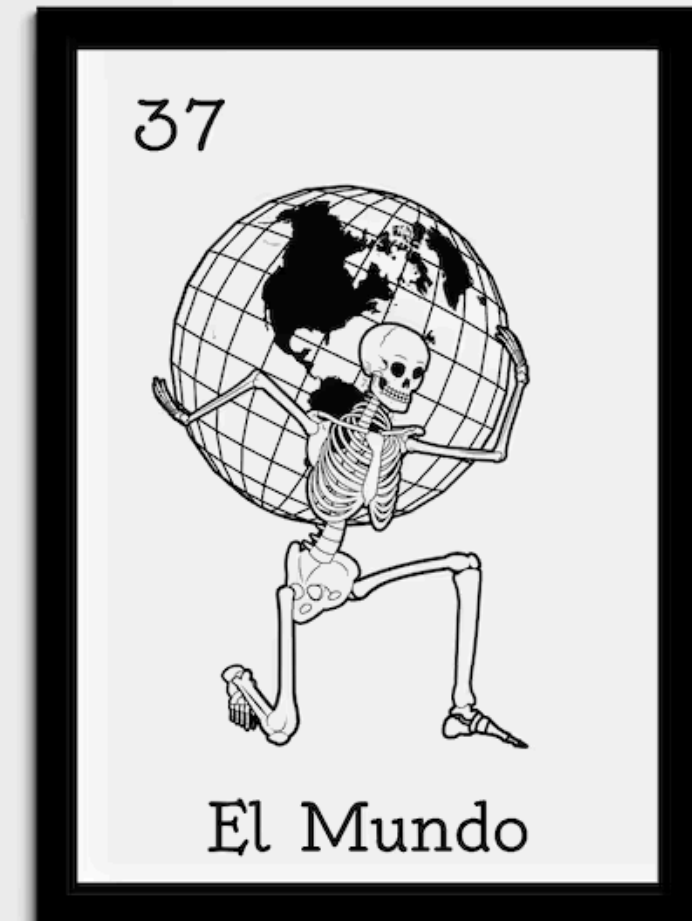
CAPÍTULO 3

PRUEBAS

Ejecución

La posición en la tabla sera totalmente aleatoria como en la vida real.

```
Opción:2
Número generado 10
El número es par. Jugador 1 Marca casilla
1.Ver tableros de loteria de jugadores
2.Tirar carta
3.Salir
Opción:1
Jugador 1:
|0||0||0||0|
|0||0||0||0|
|0||0||0||0|
|1||0||0||0|
```



CAPÍTULO 3

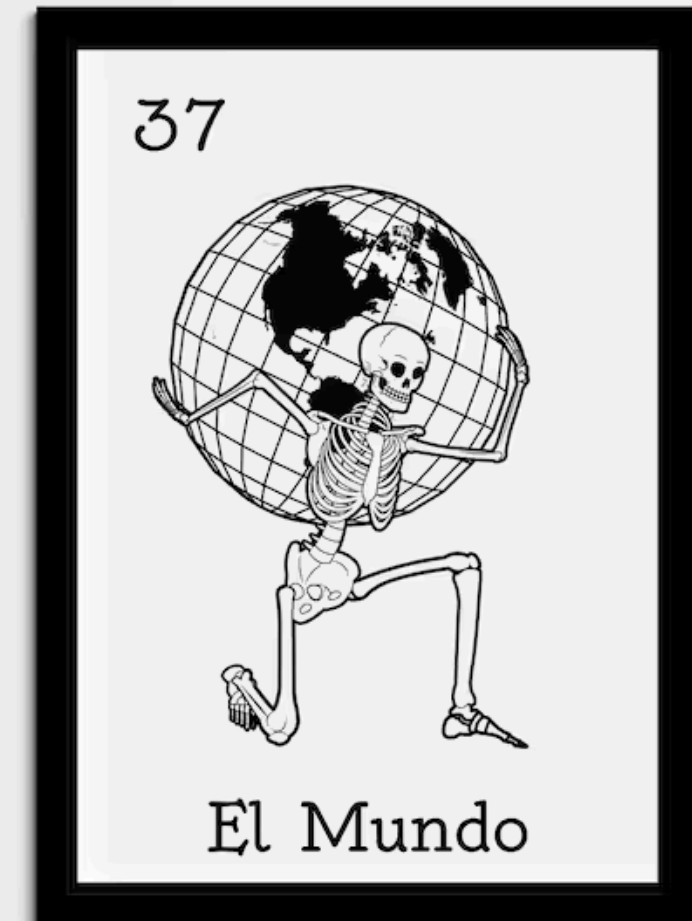
PRUEBAS

Ejecución

Al haberse llenado alguna tabla se mencionará al ganador y se mostrará el estado de las tablas.

```
El número es impar, Jugador 2 Marca casilla
Jugador 2: LOTERIAAAAAA!!!
Jugador 1:
|0||0||1||1|
|1||0||0||0|
|0||0||0||0|
|0||0||0||0|

Jugador 2:
|1||1||1||1|
|1||1||1||1|
|1||1||1||1|
|1||1||1||1|
```

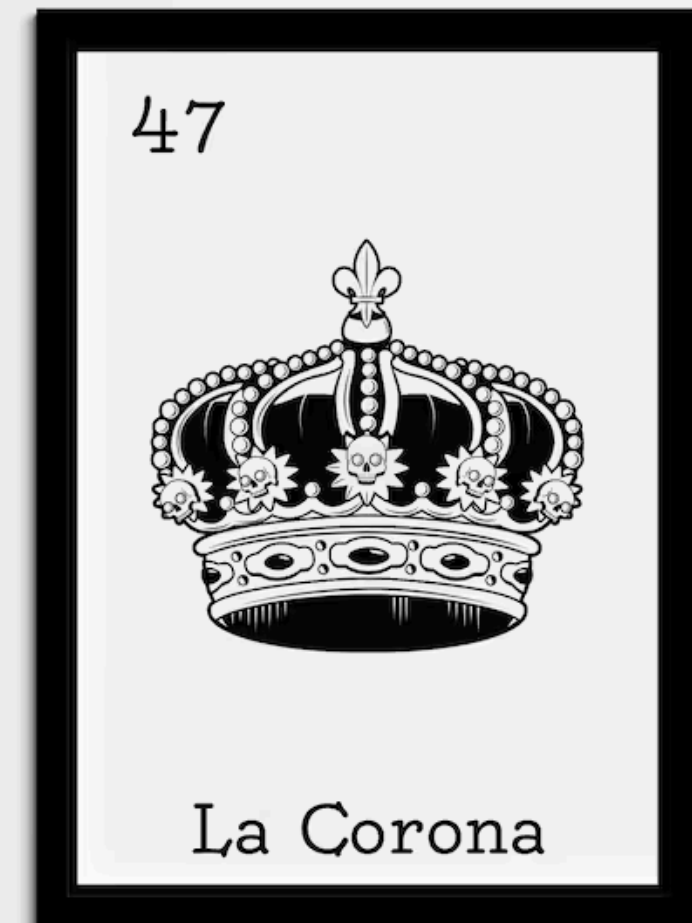


CAPÍTULO 4

CONCLUSIONES Y REFERENCIAS

Mejoras

En un futuro se puede implementar usar el nombre de cada una de las cartas de la lotería y asimismo que las tablas esten personalizadas con ciertos patrones de cartas además de que haya tablas con algunas cartas iguales inscritas en ellas.



CAPÍTULO 4

CONCLUSIONES Y REFERENCIAS

Conclusiones

El utilizar un sistema multiarchivo mejora de manera considerable el codificar en un lenguaje estructurado e implementar diversas funciones así mismo mejora la solución de problemas que se presentan haciendo más fácil encontrar los errores lógicos que tanto nos atormenta en infinidad de ocasiones.

Referencias

Kernighan, B. W., & Ritchie, D. M. (1988). The C programming language (2nd ed.). Prentice Hall.
Deitel, P., & Deitel, H. (2021). C: How to program (9th ed.). Pearson.

