

# ESCUELA COLOMBIANA DE INGENIERÍA

## PROGRAMACIÓN ORIENTADA A OBJETOS

### QuoriPOOB 2024 - 1

**Propuesta:** Juan Sebastián Ramírez, Jaider Vargas.  
**Colaboradores:** Jorge Gamboa, Laura Rodríguez

#### QUORIDOR

Quoridor es un juego de mesa para dos jugadores. Consiste en mover peones, a través de un tablero cuadrado de 9x9 casillas y colocar muros estratégicamente para bloquear el avance del oponente. Cada jugador tiene un peón y a lo sumo 10 muros, los cuales puede ubicar para obstaculizar el camino del rival hacia su objetivo. El objetivo de cada jugador es llevar su peón a la fila opuesta del tablero. Los jugadores se turnan para mover su peón o colocar un muro, buscando la forma más eficiente de avanzar mientras dificultan el progreso del contrincante. Referencia: [Quoridor](#)



#### QUORIPOOB

En el proyecto final vamos a modernizar la versión del clásico juego. Nuestra versión, **QuoriPOOB**, tendrá las siguientes novedades:

- Jugador máquina
- Límite de tiempo
- Barreras con diferentes características
- Casillas especiales

El juego tendrá solo dos jugadores.



En **QuoriPOOB** se usan las reglas internacionales de Quoridor:

- El tablero es cuadrado con casillas. (n x n)
- Ambos jugadores tienen control de un peón y n+1 barreras.
- El jugador puede mover el peón o colocar una barrera por turno.
- No está permitido obstruir el paso del otro jugador totalmente hacia el otro lado.
- Las fichas pueden saltar a la ficha contrincante siempre y cuando no haya una barrera.
- El movimiento natural de las fichas es ortogonal y unitario, sin embargo, en algunos casos cuando el paso se encuentra obstruido, la ficha puede optar por movimientos en diagonal.

Adicionalmente, ofrece las siguientes novedades:

- Se podrá especificar el tipo y cantidad de barreras que todos los jugadores tendrán durante la partida.
- Es posible jugar en tableros con casillas especiales, las cuales estarán ubicadas aleatoriamente en el tablero.

## MODOS DE JUEGO

En **QuoriPOOB** se tendrán los siguientes modos de juego:

- *Jugador vs Jugador*: En este modo se tendrán dos usuarios.
- *Jugador vs Máquina*: En esta versión uno de los dos jugadores es automático (Máquina)
- 

Las máquinas podrán tener los siguientes perfiles:

- *Principiante*: Se mueve y pone barreras al azar.
- *Intermedio*: Los movimientos de la máquina tendrán en cuenta un camino ganador del usuario y el suyo propio. De modo que compara la distancia de caminos que faltan a cada uno para llegar a la meta, si la distancia de la máquina es más corta que la del usuario decide avanzar, si no, coloca un muro.
- *Avanzado*: les una estrategia avanzada, que debe ser definida por cada equipo.

De cada jugador se debe ilustrar la siguiente información:

- **Nombre: Nombre de jugador.**
- **Color: Color de la ficha.**
- Estado del juego: Número de barreras restantes y casillas visitadas de cada tipo.

## GRADOS DE DIFICULTAD

En **QuoriPOOB** se tendrán tres modos de juego:

- **Normal: Los jugadores no tienen límite de tiempo al realizar las acciones.**
- *Contrarreloj*: Los jugadores tienen un límite de tiempo parametrizable para ejercer una acción o si no perderán el turno.
- *Cronometrado*: Se define un límite de tiempo total igual para cada jugador.

## BARRERAS

Los jugadores por turno tendrán la oportunidad de poner una barrera, las cuales pueden ser las siguientes:

### Barreras

Normal  
Temporal  
Larga  
Barreras aliadas

### Comportamiento

La barrera tiene una longitud de dos casillas.  
Después de 4 turnos la barrera desaparece. Tiene dos casillas.  
La barrera tiene una longitud de tres casillas.  
Permite pasar a través de la barrera al jugador que puso. Tiene dos casillas.

## CASILLAS

En cada turno podrán caer en una trampa o bonificación, es decir, las casillas del tablero deben poder diferenciarse de manera que indiquen la presencia de trampas o bonificaciones.

### Casillas

#### Normal

Teletransportador  
Regresar  
Turno doble

### Comportamiento

#### Campo estático normal.

Permite al usuario que esté en esta casilla moverse a su alrededor sin importar las barreras (incluye las diagonales).  
Regresa la ficha dos movimientos, si es posible.  
Roba un turno.

## REQUISITOS FUNCIONALES

La aplicación debe:

- Permitir seleccionar el tipo de oponente, persona o máquina. Si es máquina, su tipo.
- Permitir que los usuarios ingresen sus datos (color y nombre) para ser identificados.
- Permitir seleccionar el número de cada tipo de casillas especiales y de barreras.
- Generar el tablero correspondiente a la configuración definida
- Permitir definir la dificultad de juego y el tiempo asociado, si es necesario.
- Permitir mover las fichas y ejecutar las acciones respectivas.
- Definir qué tipo de barrera se quiere añadir en cada turno.
- Ilustrar la información de cada jugador
- Mostrar el tiempo restante en modos que lo requieran
- Permitir que un jugador termine el juego en cualquier momento.
- Mostrar un mensaje notificando al ganador.
- Permitir abrir y salvar el estado de un juego.

## REQUISITOS DE DISEÑO

### De extensión

- Permitir generar nuevas versiones de la aplicación que incluya otros tipos de barreras, casillas especiales y jugadores máquina.

### De visualización

- El escenario debe tener una representación gráfica adecuada que permita conocer el estado del juego.
- Los peones de cada jugador deben tener el color correspondiente
- Las barreras y casillas deben poder diferenciarse entre los diferentes tipos (formas y colores).
- La forma de seleccionar el tipo de barrera que se va a añadir al tablero debe ser simple.
- Se presentará mensajes cuando se incumpla una restricción.

### De manejo de excepciones

- Definir una clase excepción para manejar las excepciones propias.
- Incluir log de errores para los programadores.

## REQUISITOS DE ENTREGA

Revisión inicial		CAPA DE PRESENTACIÓN Boceto de la interfaz gráfica CAPA DE APLICACIÓN Diagrama de clases	A par S13: Sa 20 Abr
Versión uno Presentación	Maqueta completa Configuración de tablero Modo: Dos jugadores humanos Dificultad: Normal Barreras: dos tipos	Los entregables anteriores más CAPA PRESENTACIÓN Diagrama de clases Código CAPA DOMINIO Diagramas de secuencia Código programa Código pruebas Junit (incluir las propuestas en QuoriPOOBV1) RETROSPECTIVA	A par S14: Sa 04 May A evaluador Preliminar S15: Sa 11 May
Versión dos Persistencia	Lo anterior más Casillas: dos tipos Salvar y abrir juegos	Los entregables anteriores más PLANIFICACIÓN DE LAS CUATRO VERSIONES Detallar el objetivo de cada ciclo y los mini-ciclos que los componen Detallar si hay requisitos que quedan por fuera.	A par S16: Sa 18 May
Versión tres	La anterior más Lo propuesto para ciclo 3	Los entregables anteriores más PRUEBAS DE ACEPTACIÓN	A evaluador Inicial S17: Mc 22 May
Versión cuatro	La anterior más Lo propuesto para ciclo 4	Los entregables anteriores	A evaluador Final S18: Do 26 May
COMPETENCIA	Es requisito para participar en la competencia que el equipo se haya presentado a todas las revisiones de pares, que todas las entregas del proyecto hayan sido aprobadas y que hayan implementado y pasado las pruebas definida para la versión uno. El equipo con el mejor proyecto de todos los grupos tiene 5.0 en la nota del tercer tercio.		S18. Vi 31 May