

Instituto Tecnológico de Costa Rica

Campus Tecnológico Alajuela

Escuela de Ingeniería en Computación



Tarea Programa 1

IC-6200 Inteligencia Artificial

Especificación de agentes

Profesora:

María Auxiliadora Mora Cross

Estudiantes:

Rodolfo Cruz Vega - 2013235955

Jonathan Quesada Salas - 2020023583

ÍNDICE

1. Presenta descripción del sistema	2
2. La descripción del agente	2
3. La descripción de la función Eval	5
3.1 Heurísticas adicionales	6
4. Tablas y gráficas comparativas	6

1. Presenta descripción del sistema

El juego Nigeriano de Dara es un juego de estrategia popular en África occidental y central. El objetivo del juego es capturar las fichas del oponente y obtener más fichas que el oponente al final del juego.

El juego se juega en un tablero de juego con dos filas de 12 agujeros y dos agujeros grandes en ambos extremos, llamados "casas". Cada jugador comienza con 24 fichas, y las fichas se colocan en cada agujero, excepto en las casas.

Las reglas del juego son las siguientes:

1. Los jugadores deben decidir quién juega primero.
2. El tablero al inicio debe estar vacío.
3. En la fase 1 o fase de ubicación de fichas: Los jugadores se turnan para colocar sus fichas en las celdas vacías del tablero. (Debe existir la opción de ubicar las fichas aleatoriamente para que sea más rápido revisar la tarea). Ubicar tres en fila durante la fase 1 no es válido.
4. En la fase 2 o fase de movimiento: Los jugadores se turnarán para mover sus fichas ortogonalmente (vertical u horizontal) a una celda vacía adyacente (no es válido saltar fichas como en damas chinas). Los jugadores deben intentar hacer tres en fila con sus fichas. Solamente es permitido ubicar tres fichas seguidas, y no más. Cuatro o más fichas formadas en fila hacen un movimiento ilegal.
5. Si un jugador ubica tres fichas en fila, puede eliminar una ficha enemiga del tablero que no sea parte de un tres en fila del oponente.
6. Si un jugador ya no puede hacer tres en fila con sus piezas restantes (por ejemplo, si solo le quedan dos fichas), ese jugador pierde.
7. Si un jugador forma con éxito dos líneas de tres en fila en un movimiento, solo podrá eliminar una pieza enemiga.

2. La descripción del agente

- **Nombre del agente:**

Programa de uso del juego Dara conjunto a la funcionalidad de Minimax

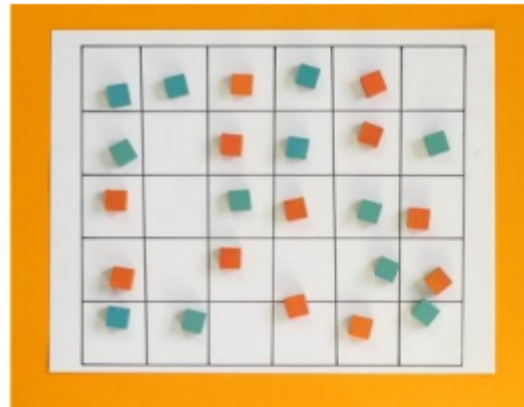
- **Descripción y objetivo del agente:**

- **Descripción:** El agente es un programa de computadora que juega automáticamente el juego Dara utilizando el algoritmo Minimax con poda α - β y heurísticas adicionales con el fin de que el tiempo de respuesta que pueda llegar a tener la inteligencia artificial sea menor a 40 segundos. El agente selecciona la mejor jugada posible en cada turno, teniendo en cuenta todas las posibles jugadas futuras y el

objetivo final de ganar el juego de manera que los tiempos de respuestas entre jugadas van a variar.

- **Entorno:** El entorno del agente es el juego Dara, que se juega en un tablero de 5×6 , donde cada jugador tiene 12 fichas. El objetivo del juego es capturar las fichas del oponente moviendo una ficha propia a una casilla adyacente vacía o capturando una ficha del oponente al saltar sobre ella. El juego termina cuando uno de los jugadores no puede realizar una jugada legal..
- **Percepción:** El agente percibe el estado actual del tablero, incluyendo la posición de las fichas de ambos jugadores y las posibles jugadas legales disponibles en el momento. Además, el agente recibe información sobre la jugada realizada por el jugador humano
 - ubicación de la ficha y posibles resultados
 - [(A,B), [[Jugadas]]]
Jugadas: Lista de listas donde se almacena las posibles jugadas
- **Acción:** El agente realiza jugadas mediante la selección de una ficha y la selección de una casilla adyacente vacía o la selección de una ficha del oponente para capturar. El agente selecciona la mejor jugada posible en cada turno utilizando el algoritmo Minimax con poda α - β y heurísticas adicionales.
 - Jugadas: mover(Izquierda, Derecha, Arriba, Abajo), validador _3_ línea([Posición fichas] [Posibles jugadas]), comer(QuitarFicha), no comer(NoOp)
- **Medida de desempeño**
 - La medida de desempeño del agente es el número de juegos ganados, perdidos o empatados frente al jugador humano. El objetivo del agente es maximizar el número de juegos ganados.
- **Objetivo del agente:** El objetivo del agente es maximizar el número de juegos ganados frente al jugador humano. Para lograr este objetivo, el agente utiliza el algoritmo Minimax con poda α - β y heurísticas adicionales para seleccionar la mejor jugada posible en cada turno.
- **Función del agente:** La función del agente es jugar el juego Dara de forma autónoma, seleccionando la mejor jugada posible en cada turno y tratando de ganar el juego.

DARA
 abstract strategy
 game from Nigeria
 3 in a row
 with a twist



- El entorno de la tarea (PEAS): la medida de desempeño, el ambiente, los actuadores y los sensores:
 - Medida de desempeño:
 - La medida de desempeño del agente es el número de juegos ganados, perdidos o empatados frente al jugador humano. El objetivo del agente es maximizar el número de juegos ganados.
 - Ambiente:
 - El ambiente del agente es el juego Dara. El juego se juega en un tablero de 5 × 6, donde cada jugador tiene 12 fichas. El objetivo del juego es capturar las fichas del oponente moviendo una ficha propia a una casilla adyacente vacía o capturando una ficha del oponente al saltar sobre ella. El juego termina cuando uno de los jugadores no puede realizar una jugada legal
 - Los actuadores:
 - El agente realiza jugadas mediante la selección de una ficha y la selección de una casilla adyacente vacía o la selección de una ficha del oponente para capturar
 - Los sensores:
 - El agente percibe el estado actual del tablero, incluyendo la posición de las fichas de ambos jugadores y las posibles jugadas legales disponibles en el momento. Además, el agente recibe información sobre la jugada realizada por el jugador humano.

Tipo de agente	El agente es un jugador autónomo que utiliza el algoritmo Minimax con poda α - β y heurísticas adicionales para seleccionar la mejor jugada posible en el juego Dara.
----------------	--

Medida de desempeño	La medida de desempeño del agente es el número de juegos ganados, perdidos o empatados frente al jugador humano. El objetivo del agente es maximizar el número de juegos ganados.
Ambiente	El ambiente del agente es el juego Dara. El juego se juega en un tablero de 5×6 , donde cada jugador tiene 12 fichas. El objetivo del juego es capturar las fichas del oponente moviendo una ficha propia a una casilla adyacente vacía o capturando una ficha del oponente al saltar sobre ella. El juego termina cuando uno de los jugadores no puede realizar una jugada legal.
Actuadores	El agente realiza jugadas mediante la selección de una ficha y la selección de una casilla adyacente vacía o la selección de una ficha del oponente para capturar.
Sensores	El agente percibe el estado actual del tablero, incluyendo la posición de las fichas de ambos jugadores y las posibles jugadas legales disponibles en el momento. Además, el agente recibe información sobre la jugada realizada por el jugador humano.

3. La descripción de la función Eval

La función evaluate-board es una función heurística que calcula un valor numérico para un estado de tablero en el juego de Dara, en función de ciertos criterios. Estos criterios se utilizan para ayudar a una inteligencia artificial a tomar decisiones sobre qué movimiento realizar.

Las heurísticas que se aplican son las siguientes:

- player1-pieces y player2-pieces se utilizan para contar el número de piezas que tiene cada jugador en el tablero.
- total-pieces es la suma total de piezas en el tablero.
- Diferencia en el número de piezas: Esta heurística evalúa la diferencia en el número de piezas que tiene cada jugador en el tablero. Si el jugador 2 tiene más piezas que el jugador 1, la heurística utiliza una condición para

determinar qué valor devolver. Si la diferencia es de 2 piezas, devuelve un valor de 49. Si la diferencia es de 1 pieza, devuelve un valor de 48. Si la diferencia es mayor, devuelve un valor de 50.

- **lost?** se utiliza para determinar si alguno de los jugadores ha perdido.
- **cond** es una estructura condicional que utiliza diferentes criterios para asignar valores a las posibles situaciones del tablero. Por ejemplo, si el jugador 1 ha perdido, se asigna un valor de -10000 a esa situación. Si el jugador 2 ha perdido, se asigna un valor de 10000. Si el jugador 1 tiene más piezas que el jugador 2, se asigna un valor de 5000, y si el jugador 2 tiene más piezas que el jugador 1, se asigna un valor de -5000.

3.1 Heurísticas adicionales

Para las heurísticas adicionales para el juego Dara, que se utilizan para evaluar el estado actual del tablero del juego y determinar la mejor jugada a realizar por el jugador en turno. Esas se utilizaron para realizar pruebas y encontrar el mejor balance entre jugabilidad y duración. Finalmente se incluyeron solo las de evaluar si el jugador pierde y diferencia de fichas. A continuación se explican las diferentes heurísticas utilizadas:

- **Ratio de piezas:** Esta heurística evalúa la proporción de piezas que tiene cada jugador en el tablero. Si el número de piezas del jugador 1 es mayor que el número de piezas del jugador 2, la heurística devuelve un valor positivo de 30. Si el número de piezas del jugador 1 es menor que el número de piezas del jugador 2, la heurística devuelve un valor negativo de -30.
- **Distancia con el centro:** Se calcula la distancia promedio con el centro de las piezas de cada jugador, se favorece aquellos estados en los que el jugador está controlando el centro.
- **Diferencia en el número de movimientos posibles:** Esta heurística evalúa la diferencia en el número de movimientos posibles que tiene cada jugador en el tablero. Si el jugador 2 tiene más movimientos posibles que el jugador 1, la heurística devuelve un valor positivo de 60. Si el jugador 1 tiene más movimientos posibles que el jugador 2, la heurística devuelve un valor negativo de -60.
- **Elección de pieza a remover:** Se elige la pieza para remover en caso de lograr hacer 3 en línea buscando la ficha del oponente que tenga la posibilidad de formar 3 en línea revisando las fichas cercanas.

Cada una de estas heurísticas devuelve un valor que se suma al puntaje total del estado del tablero. La heurística final se calcula sumando todos los valores devueltos por las diferentes heurísticas.

4. Tablas y gráficas comparativas

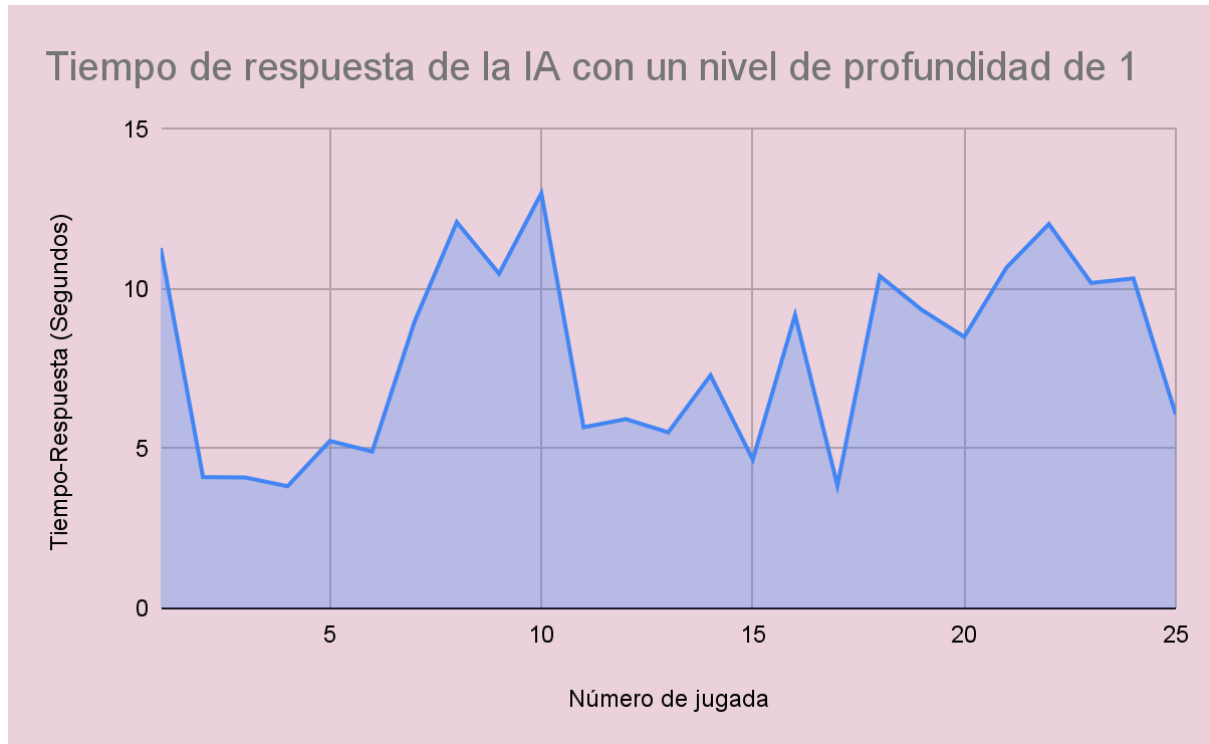
Con lo que respecta a este programa que se desarrolló previamente se realizaron pruebas de rendimiento para poder comprobar la eficiencia y eficacia del programa de Dara según el nivel de profundidad, a continuación se puede observar una tabla con tiempos de duración de ejecución que le tomó al programa al seguir con la siguiente jugada con un valor en la profundidad del árbol en 1:

Tabla 1
Tiempo de respuesta profundidad 1

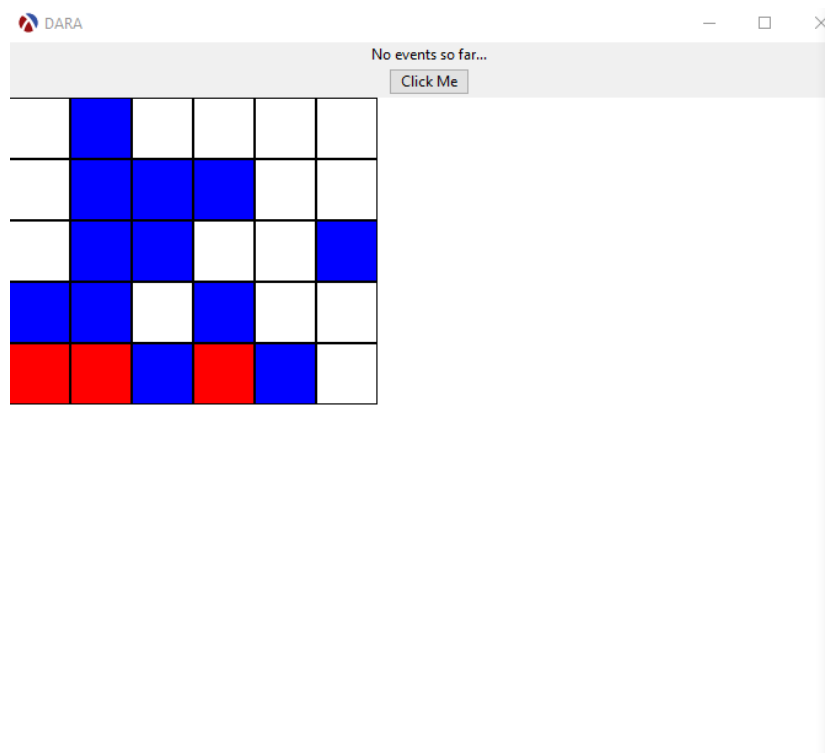
Con nivel de profundidad 1	
Jugada	Tiempo de respuesta (Segundos)
1	11.28386304
2	4.110212646
3	4.097572754
4	3.823381348
5	5.242339111
6	4.910087158
7	8.972621826
8	12.09291235
9	10.48051953
10	12.99976465
11	5.671229736
12	5.925956055
13	5.511477295
14	7.297779053
15	4.657677734
16	9.186103516
17	3.849444336
18	10.40750854
19	9.340875244
20	8.501019775
21	10.67287988
22	12.03448682
23	10.19081006
24	10.33059937
25	6.073652588
PROMEDIO	7.906590977

A continuación para verlo de una manera más gráfica se adjuntará gráficas para mayor visualización:

Gráfica 1
Tiempo de Respuesta profundidad 1



Gráfica 1
Tiempo de Respuesta profundidad 1



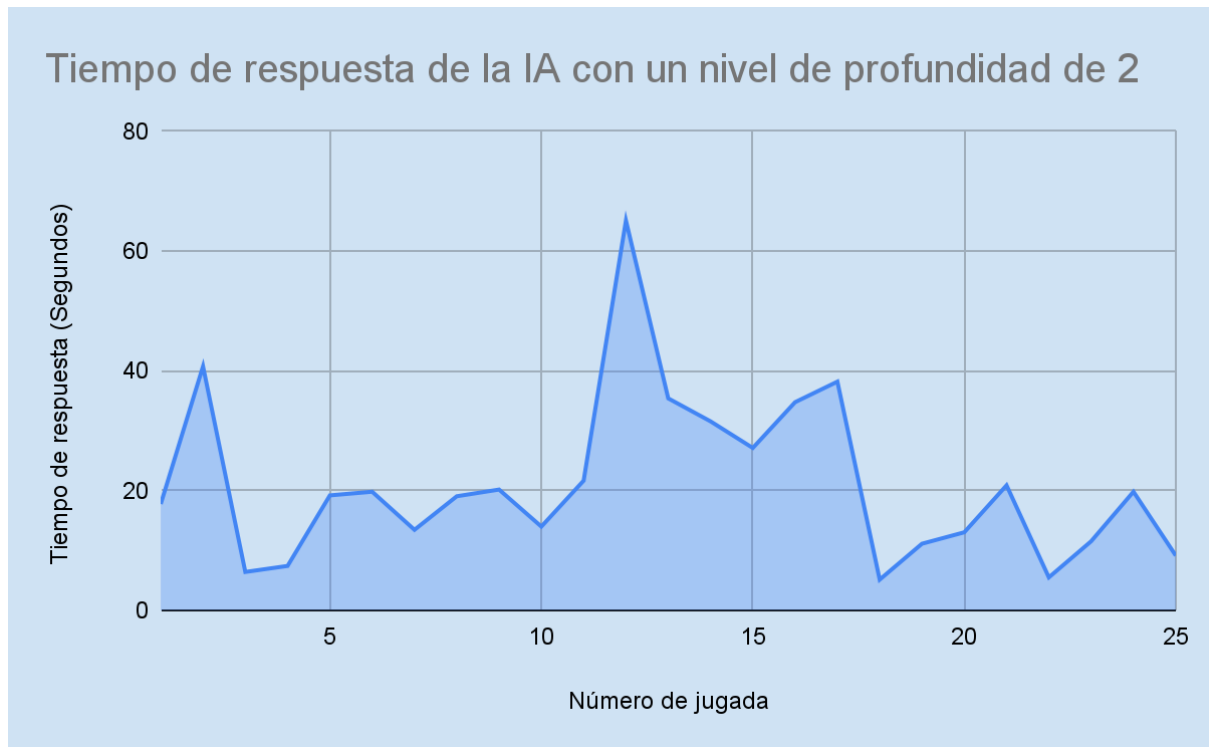
Posteriormente a las gráficas presentadas es necesario poder realizar el mismo procedimiento para calcular los tiempo pero en este caso para un nivel de profundidad al árbol de 2:

Tabla 2
Tiempo de respuesta profundidad 2

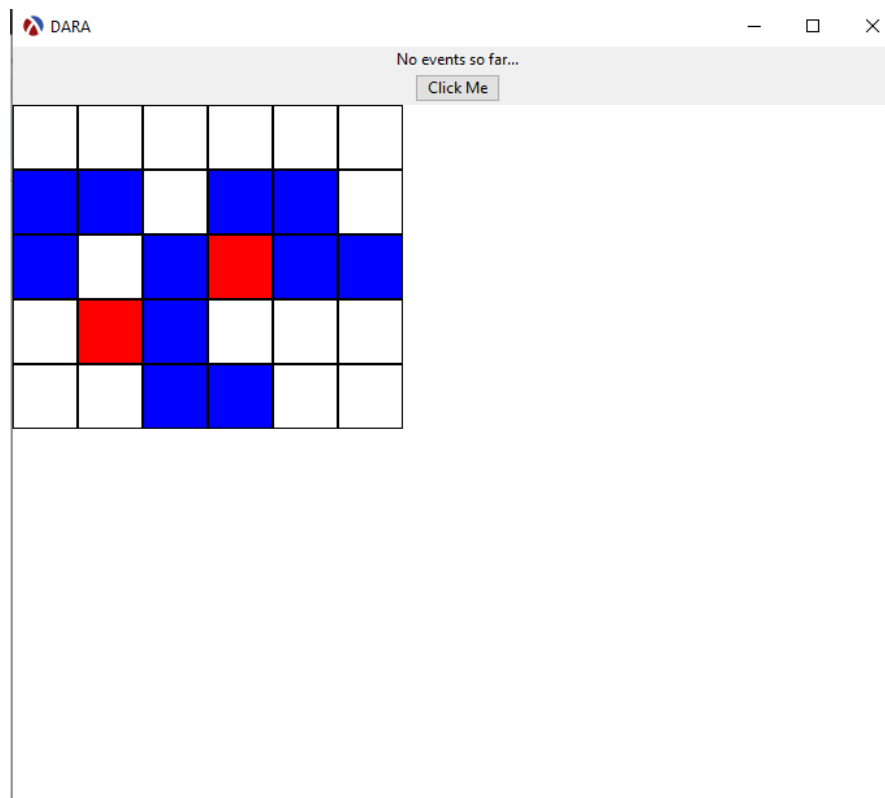
Con nivel de profundidad 2	
Jugada	Tiempo de respuesta
1	17.7600188
2	40.74691895
3	6.409569336
4	7.416606445
5	19.19725122
6	19.80713867
7	13.4459209
8	19.05124902
9	20.16964063
10	13.98684375
11	21.66018872
12	65.10019507
13	35.39838184
14	31.56372632
15	27.11982153
16	34.76330371
17	38.19332495
18	5.129993408
19	11.12415234
20	13.03015479
21	20.87654297
22	5.500568359
23	11.53374878
24	19.81039478
25	9.148035156
PROMEDIO	21.11774762

A continuación para verlo de una manera más gráfica se adjuntará gráficas para mayor visualización:

Gráfica 2
Tiempo de Respuesta profundidad 2



Gráfica 2
Tiempo de Respuesta profundidad 2



Y por último se deberán de realizar las mismas pruebas para un nivel de profundidad de valor 3, para de esta manera poder bajar más en el árbol de decisión

por medio de las heurísticas implementadas de manera que se puede apreciar que el tiempo va a incrementar de una manera mayor a comparación a los niveles de profundidad anteriores:

Tabla 3
Tiempo de respuesta profundidad 3

Con nivel de profundidad 3	
Jugada	Tiempo de respuesta (Segundos)
1	163.7333975
2	131.9380393
3	187.0839421
4	145.8333291
5	210.9623044
6	136.9972754
7	136.9972754
8	148.6259731
9	200.0530803
10	159.5144011
11	98.34183228
12	173.0628835
13	620.9552056
14	114.1845713
15	69.74894116
16	66.13135156
17	131.6429884
18	61.74761597
19	285.6210859
20	353.2814329
21	159.125292
22	227.549812
23	173.0628835
24	86.95115615
25	136.1539762
PROMEDIO	175.1720019

A continuación para verlo de una manera gráfica se adjuntará gráficas para mayor visualización:

Gráfica 3

Tiempo de Respuesta profundidad 3

Tiempo de respuesta de la IA con un nivel de profundidad de 3

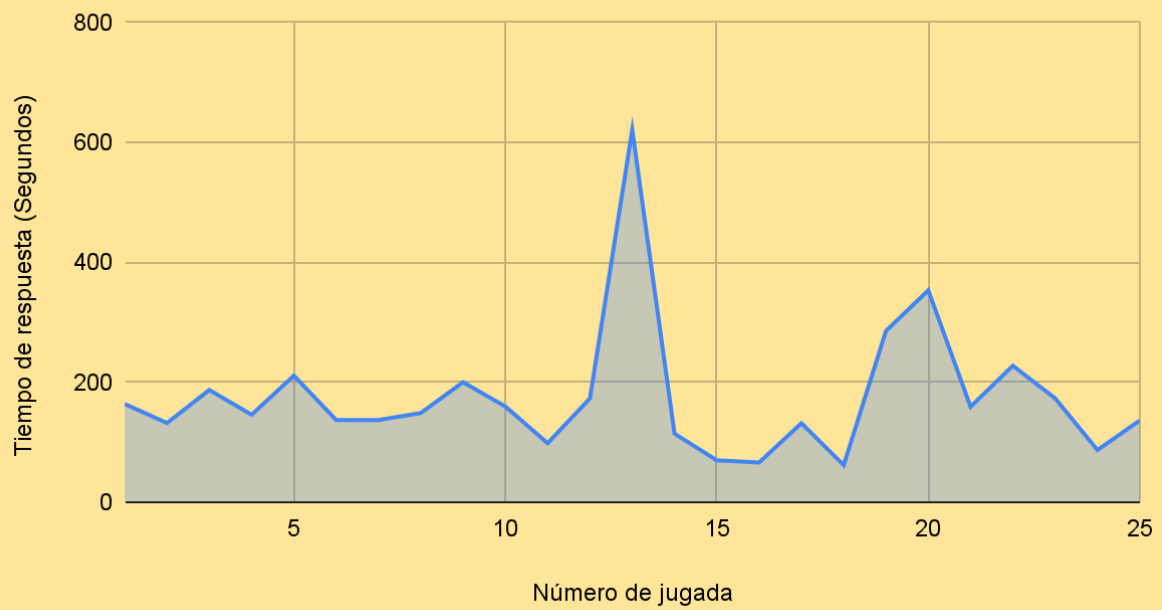
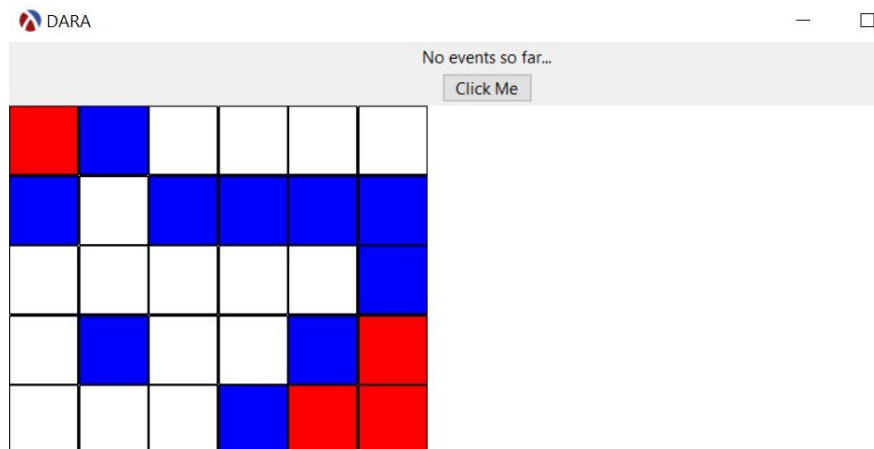


Imagen 3

Estado del tablero final con profundidad 3



Resumen de resultados de evaluacion de tiempo de respuesta por profundidad:

Tabla 4
Promedio, max, min, y desviación estándar del tiempo de respuesta

Tiempo de respuesta			
	Profundidad 1	Profundidad 2	Profundidad 3
Promedio	7.906590977	21.11774762	175.1720019
Max	12.99976465	65.10019507	620.9552056
Min	3.823381348	5.129993408	61.74761597
D. estándar	0.07012446078	0.5330521926	6.98430075

Gráfica
Tiempo de respuesta por segunda según nivel de profundidad

Tiempo (en segundos) de respuesta por jugada

