

Proyecto Programado 1

Objetivo general: crear un juego de computadora que juegue contra un adversario humano al Juego Real de Ur.

Objetivo didáctico general: los estudiantes consolidarán su conocimiento en los algoritmos de búsqueda sin información en un espacio de estados como estrategia de resolver problemas.

Temas del programa cubiertos por el proyecto: Búsqueda sin información.

Objetivos específicos:

1. Implementación de búsqueda iterativa.
2. Implementación de un programa de juego contra un adversario, en un juego de suma cero con conocimiento completo.
3. Implementación del algoritmo minimax con poda $\alpha\beta$.
4. Implementación de una interfaz gráfica de juego agradable para el usuario.

Descripción del juego:

El juego Real de Ur es un juego muy antiguo, y no se sabe todo acerca de él, incluyendo su nombre. Puesto que no es ampliamente conocido hoy en día, y ya que no se conoce un juego definitivo de reglas, hay varias conjeturas diferentes acerca de cómo se pudo haber jugado. Para este trabajo hemos escogido el modo recomendado más sencillo.

El juego Real de Ur pertenece a la familia de los llamados **juegos de carrera**, como el juego de la Oca, el Backgammon o Jumanji; esto significa que las piezas no se capturan, sacándolas del tablero, sino que el juego consiste en sacar las propias piezas del tablero; el primero que saca todas sus piezas gana. Este es un juego para dos jugadores, sobre un tablero de 20 casillas que se muestra en la figura 1.

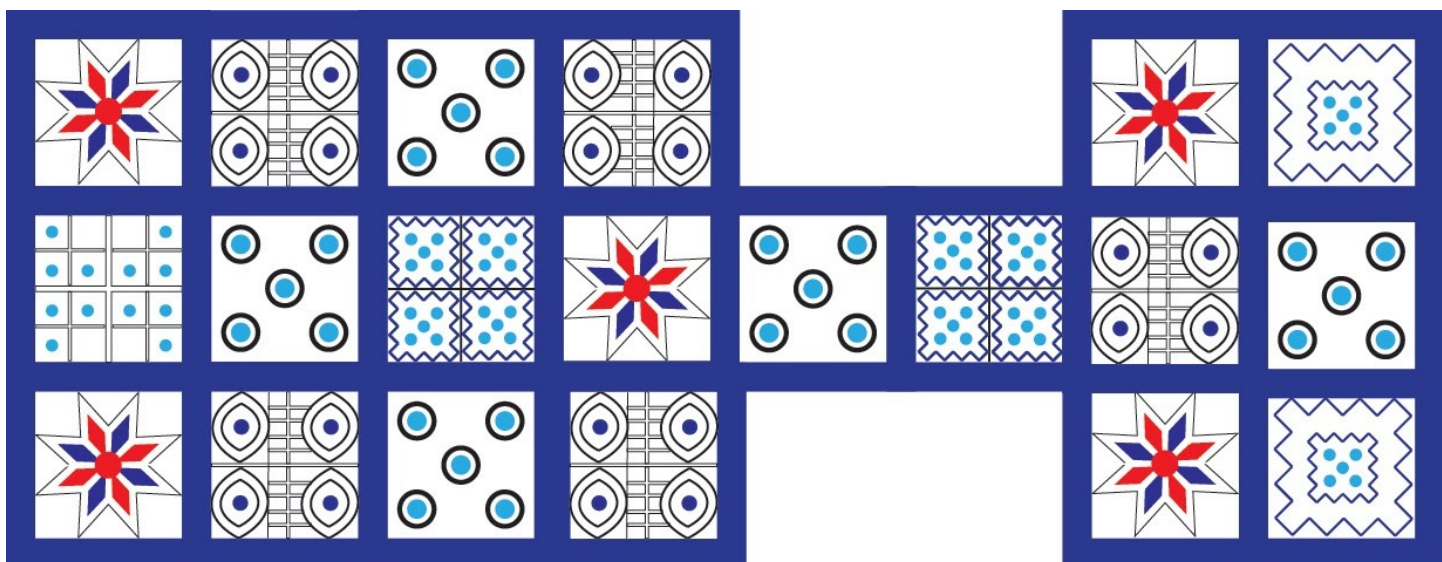


Figura 1. Tablero del Juego Real de Ur, desde el punto de vista de “blanco”

Reglas del juego:

1. Cada jugador tiene siete “tokens”, que son piezas circulares que tienen una de sus caras lisa y la otra con 5 puntos. Los tokens de uno de los jugadores son negros y los del otro jugador son blancos.
2. La selección del jugador que inicia es arbitraria.
3. Hay dos casillas que se designan: casa del blanco y casa del negro, como se muestra en la figura 2.

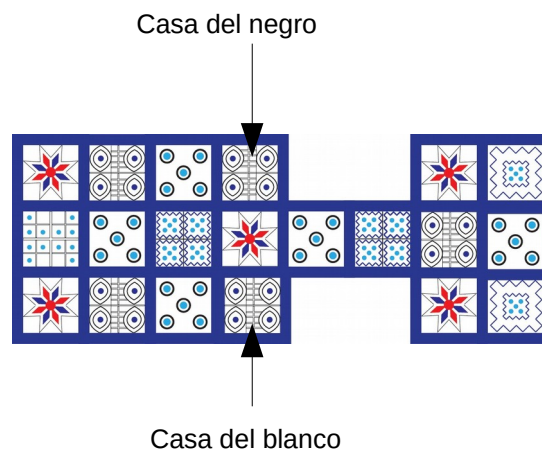


Figura 2. Ubicación de las casas de cada jugador

4. Cada jugador se ubica del lado del tablero en que está su casa
5. El objeto del juego es hacer entrar en el tablero la piezas, una a la vez, a partir de su propia casa. Cada una de las piezas de cada jugador tiene que hacer un recorrido fijo, que se muestra en las figuras 3 y 4.

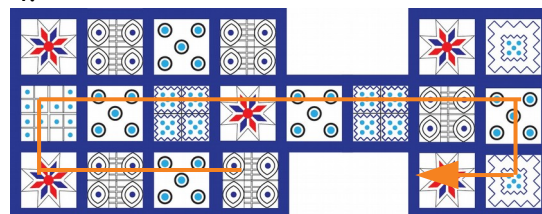


Figura 3. Recorrido de las piezas blancas

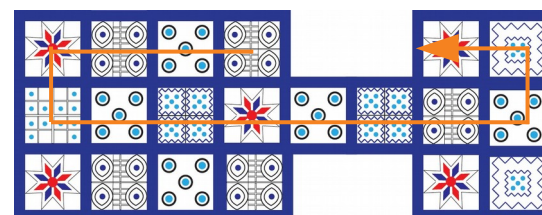


Figura 4. Recorrido de las piezas negras

6. El juego original constaba de tres “dados” tetraédricos. Un tetraedro tiene cuatro esquinas, y los de este juego se pintaban dos negras y dos blancas, y cada esquina blanca tenía el significado de “1”. Entonces, al arrojar los dados y sumar las esquinas blancas se puede

- generar cualquier número entre 0 y 3. El sistema de numeración sumerio desconocía el “0”, por lo que la ausencia de esquinas blancas se interpretaba como 4. Si una pieza se está ingresando al tablero entonces la casa cuenta por 1. Las probabilidades de los números de 1 a 4 son diferentes (es más probable que salga un 1 o un 2).
7. La cuenta que salga en los dados se debe aplicar a una sola pieza propia en el tablero, pero el jugador elige a cuál la aplica.
 8. Para que una pieza pueda salir del tablero se debe dar una cuenta exacta de caer en la última casilla más uno.
 9. Una casilla solo podrá contener un token.
 10. Si se cae sobre una casilla ocupada por una pieza del adversario entonces se ocupa esa casilla y la del adversario sale del tablero y deberá volver a entrar por su casa; hay una excepción a esta regla, que se verá más adelante.
 11. Las casillas marcadas con una roseta tienen propiedades especiales. En primer lugar, el jugador que cae en ella tiene un segundo turno. En segundo lugar, una pieza que se encuentre en una roseta no puede ser desplazada por una pieza del adversario.



Interfaz de usuario:

El programa deberá mostrar el tablero en una ventana gráfica, similar al que se muestra en la figura

1. Se mostrará la salida de los dados y el resultado del lanzamiento. Habrá un cuadro de diálogo que le permita al usuario dictar su jugada, aunque también se podrá usar el mouse para mover las piezas.

Detalles de trabajo

1. Se podrá trabajar en equipos de hasta 4 personas.
2. La revisión será el 29 de septiembre de 2020, de 9:00am a 6:30pm
3. En un documento aparte se hará una descripción matemática de la función de evaluación del tablero y del algoritmo de poda empleado.
4. La máquina debería ser capaz de sacar una de sus piezas antes que su oponente humano.