

Hive

Nadia González Fernández C-412

Luis Alejandro Lara Rojas C-412

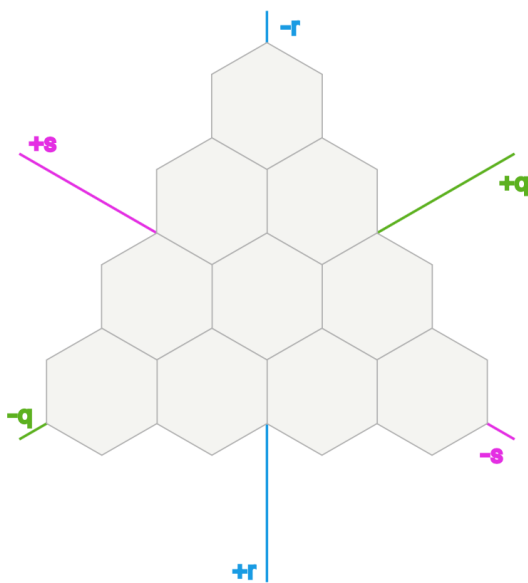
Introducción:

El proyecto implementa el juego Hive. En este se puede jugar entre jugadores o con máquinas. Para su desarrollo se utilizó un tablero de 3 ejes con las coordenadas (Q, R, S). La simulación de jugadores se realizó mediante un algoritmo de inteligencia artificial. Para el almacenamiento del estado del juego se utilizó la base de datos de prolog, con la cual se interactúa durante la ejecución del juego.

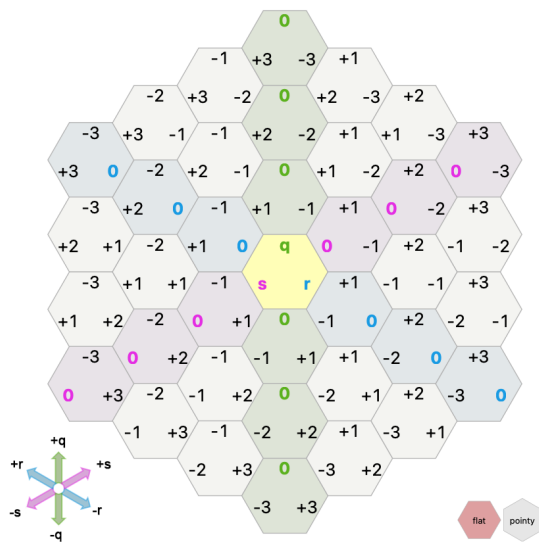
Estructura del Juego

Tablero:

Para representar el tablero hexagonal del juego se utilizan 3 ejes primarios, en vez de dos como se utiliza en grids cuadrados.



Las coordenadas son:



Ejecución del Juego

Para inicializar hive debe en consola ejecutar el archivo `main.pl` y ejecutar el predicado `start_game`:

```
swipl
```

```
[main].
```

```
start_game.
```

Se mostrarán a continuación las reglas del juego y dará a escoger el modo de juego que desee.

```
CHOOSE A GAME MODE:
1. WHITE PLAYER vs BLACK MACHINE
2. BLACK PLAYER vs WHITE MACHINE
3. PLAYER vs PLAYER
4. MACHINE vs MACHINE
```

Estos son:

- Jugador Blanco Vs. Máquina Negra.
- Jugador Negro Vs. Máquina Blanca.
- Jugador Vs. Jugador.
- Máquina Vs. Máquina.

Para iniciar la partida introduzca el número del modo deseado. Para iniciar en modo Jugador Blanco Vs. Máquina Negra debe escribir:

```
"1".
```

Las piezas se muestran de la siguiente forma:

```
[COLOR TYPE Q R S]
```

Ejemplo:

```
[W Q 0 0 0]
```

Lo anterior representa una pieza reina blanca en la posición (0, 0, 0).

La siguiente tabla contiene la leyenda del juego:

Colores

Blanco	W
--------	---

Negro	B
-------	---

Insecto	Init	Move
Abeja Reina	queen	Q
Escarabajo	beetle	B
Saltamontes	grasshopper	G
Araña	spider	S
Hormiga Soldado	ant	A
Mariquita	ladybug	L
Mosquito	mosquito	M
Bicho Bola	pillbug	P

Movimientos

Para inicializar una pieza:

```
"init bug Q R S".
```

Ejemplo:

```
"init queen 0 0 0".
```

Para mover una pieza ya inicializada

```
"move Q R S to NewQ NewR NewS".
```

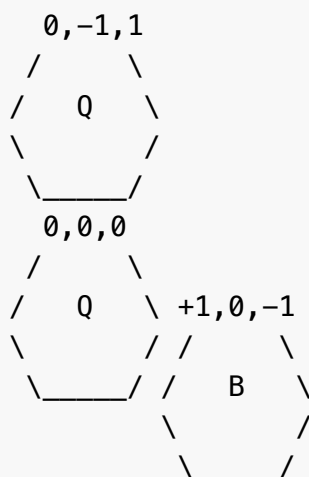
Ejemplo:

```
"move 0 0 0 to 0 -1 1".
```

El tablero se representa de la siguiente forma:

```
Pieces:
[W Q 0 0 0]
[B Q 0 -1 1]
[W B 1 0 -1]
```

En este caso el tablero sería:



Nota: Siempre puede consultar la imagen del tablero presentada anteriormente para cualquier duda sobre las posiciones.

Otros comandos

Para salir del juego se utiliza el comando:

```
"exit".
```

Para mostrar las instrucciones se utiliza el comando:

```
"instructions".
```

Nota: Todos los comandos que se inserten deben estar entre comillas dobles (") y terminar en punto (.).

Inteligencia Artificial

El algoritmo utilizado es AlphaBeta, de tipo MiniMax, el cual explora el árbol de posibles movimientos para calcular cuál es el mejor utilizando alguna heurística.

La heurística utilizada en este proyecto es $EM + EQ - MQ$, donde:

- Cantidad de movimientos posibles del adversario: EM
- Cantidad de espacios adyacentes a la Reina Enemiga libres: EQ
- Cantidad de espacios adyacentes a la Reina Aliada libres: MQ

Para reducir el tiempo de ejecución del algoritmo, se establece una profundidad máxima en la cual se va a podar el árbol de posibles jugadas.

Esta profundidad está igualada a 1 en el archivo `main.pl` en la línea 78. Este valor puede ser cambiado en caso de querer una mayor profundidad.