

Informe Proyecto Segunda Etapa

Adrover Ramiro, LU: 125925

Sabater Joaquín, LU: 125817

Lógica para Ciencias de la Computación

Trabajo Realizado:

Para la segunda entrega del proyecto extenderemos la el enunciado de la primera agregando la posibilidad de que el jugador solicite ayuda para ganar el juego a través de 2 métodos que generan una secuencia de jugadas que a partir de la grilla actual acumulan la mayor cantidad de celdas dependiendo de la cantidad de movimientos a generar por el jugador.

Modificaciones desde la primera parte:

Se agregó a la pantalla de juego lo siguiente:

- Un campo de texto en donde solo se permiten números positivos
- Un botón de ayuda con el texto "Help" que permitirá al jugador solicitar ayuda para la cantidad de movimientos indicada en el campo de texto dicha ayuda corresponde al algoritmo denominado optimal.
- Un segundo botón de ayuda con el texto "Help greedy" que permitirá al jugador solicitar ayuda para la cantidad de movimientos indicada en el campo de texto, dicha ayuda corresponde al algoritmo denominado greedy.

Modificaciones a nivel de código en Prolog:

Se removi6 el método winnerShell/6 y ahora la detección de juego terminado se hace con las cantidad de celdas obtenidas luego de hacer un flick.

- Se hace uso del código para obtener AdyacentesC* provisto por la cátedra.

|Predicados Principales de Prolog:

- combsShell/8: predicado que devolverá la mejor combinación para una grilla, con su posición inicial y la cantidad actual de capturados.
- allCombs/7: para la grilla que recibe, retorna sus el siguiente nivel de combinaciones con sus secuencias y grillas resultantes asociadas.
- combinations/5: predicado que descompone las ternas de secuencia,cantidad de capturadas y grillas y además hace uso de allCombs/7 para obtener los siguientes niveles.
- greedy/8: predicado alternativo para obtener la mejor secuencia de movimientos para una cantidad especificada.
- getBestMove/7: predicado que para una grilla, cantidad de capturadas obtiene el movimiento que captura más celdas en esa para esa grilla.

Observaciones:

Para la segunda entrega del proyecto

- Se hace uso del operador de corte “Cut” (!) definido en la teoría.
- El conjunto de celdas llamado AdyacentesC está definido como la celda seleccionada como inicial al comienzo de la partida o cualquier celda adyacente a la inicial y del mismo color o cualquier celda adyacente a una celda adyacente a la inicial y del mismo color.

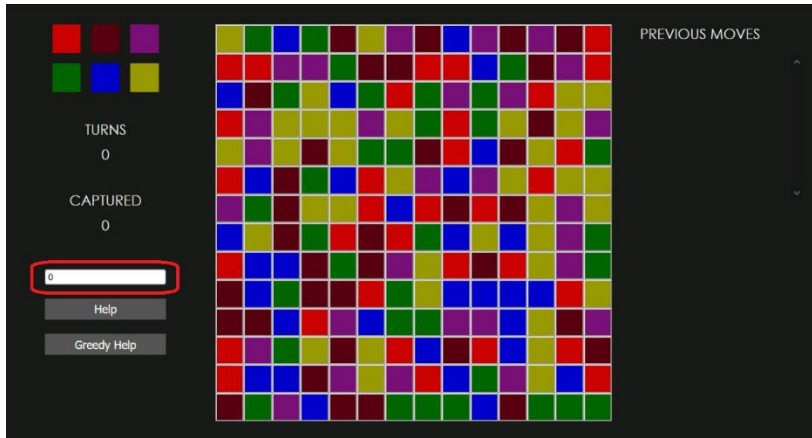
Predicados predefinidos

Para la implementación de este proyecto se utilizaron los siguientes predicados predefinidos del prolog.

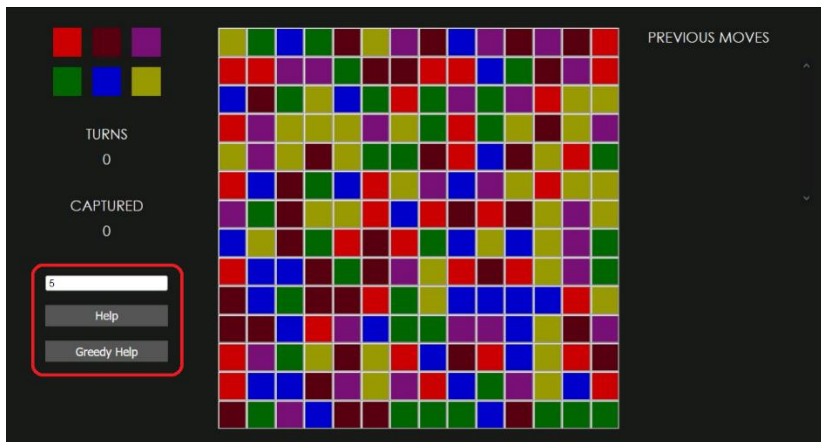
- Findall/3
- Append/3
- Member/2
- Length/2

Cómo Solicitar Ayuda (para el Manual):

- Ingresar en el campo de texto la cantidad de movimientos a predecir.



- Elegir uno de los botones de ayuda y apretarlo



- Observar en el apartado “Recommended Moves” la secuencia que capturara la mayor cantidad de celdas para la cantidad de movimientos indicada.

