

Reporte Segundo Proyecto Programacion

MATCOM
2022 – 2023

Implementacion de un juego de Domino

Adrian Hernandez Santos – C211
Alejandro Alvarez Lamazares – C212

Sobre el juego:

Nuestra implementacion del juego de domino esta basada en el juego clasico, tanto doble 9, como doble 7, para la implementacion inicial, las reglas, etc. Partiendo de ahi pudimos determinar distintos conceptos base para modificar y crear un juego mas extensible que su variante clasica. Apoyandonos en los conceptos basicos de Programacion Orientada a Objetos aprendidos en el curso de Programacion, tuvimos las herramientas necesarias para crear un proyecto base en el cual cualquiera en un futuro puede llegar e implementar nuevas funcionalidades con un minimo de codigo en la mayoria de los casos, siempre que este definida una abstraccion para dicho elemento a extender.

Aspectos clave:

- Fichas
- Jugadores
- Condicion de Ganado
- Ganador

Fichas:

Las fichas en el domino son la parte mas importante, puesto que en ellas se basa todo el sistema de puntaje, los posibles movimientos y demas funcionalidades que permiten el correcto funcionamiento del juego. Para el nuestro, decidimos implementar 3 variantes:

DominoToken:

Representa la ficha estandar del juego de domino. Su calculo de valor esta dado por la suma de los valores de cada componente (izquierda y derecha) como en el juego normal. Ademas de poseer un sistema para generar las fichas dada una numeracion de juego (doble 9 por ejemplo), el cual es heredado a cada una de las fichas hijas para que tenga la misma funcionalidad, dependiendo del tipo de ficha en cuestion.

SixUnvaluableDominoToken:

Representa una ficha en la cual el valor 6 esta penalizado. Esto quiere decir que si en alguna de sus componentes aparece un 6, el valor del mismo automaticamente pasa a ser 0, o sea, no tiene valor.

DoubledValueDominoToken:

Representa una ficha en la cual el valor de cada una de sus componentes es el doble de lo que tuviera si fuera una implementacion estandar de dicha ficha. Por ejemplo, el doble 3, al ser sus 2 componentes 3 y 3, se convierte (representativamente) en un doble 6, puesto que el valor de cada una se duplica, pero sigue siendo un 3 a la hora de jugarla.

Jugadores:

El jugador es una parte clave en el juego de domino. Sin el el propio concepto de juego no tendria sentido. La funcionalidad basica del jugador seria jugar una ficha en su turno, dado un estado del juego, dependiendo de su implementacion. Dicha funcionalidad tiene 2 momentos clave, por una parte la ficha inicial, si el jugador es el que comienza la partida, y por otro, jugar una ficha en su turno, despues de una cierta cantidad de jugadas. Por su importancia en el propio juego fue que decidimos hacer varias implementaciones de estos, en especifico 4:

RandomDominoPlayer:

Un jugador que seleccionara una ficha aleatoria para jugar de las posibles jugadas que pueda hacer, basado en el sistema RNG de .NET.

GreedyDominoPlayer:

Un jugador que jugara siempre la ficha con mayor valor de entre las posibles jugadas que tenga en mano, independientemente de la forma de calcular el valor del tipo de ficha que se utilice en el juego.

HumanDominoPlayer:

Implementacion de un jugador humano, se le pedira al usuario que dado un estado del juego y una mano de posibles fichas, realice una jugada en su turno, con una entrada en la terminal de la consola del juego.

DataDominoPlayer:

Jugador que basara su jugada primeramente seleccionando de una data (numero con mayor ocurrencia entre sus fichas), y posteriormente, en caso de no poder jugar por su data, jugara la primera ficha que sea posible.

Condicion de Ganado:

La condicion de ganado en el domino determina cuando un juego dado debe terminar. Independientemente del juego que se juegue, luego de cada jugada de un jugador, debe verificarse el estado de la partida, para decidir si se debe continuar o se debe terminar dicho juego. Hemos realizado 2 implementaciones para esta funcionalidad:

StandardWinCondition:

Condicion de ganado estandar, o sea, las reglas basicas para que una partida termine: que nadie pueda hacer una jugada o que algun jugador se quede sin fichas en la mano.

FourRoundsWinCondition:

Condicion de ganado que consiste en que el juego finaliza luego de 4 rondas, o cuando un jugador se queda sin fichas. Una ronda esta definida como la sucesion de jugadas desde un jugador a el mismo nuevamente, o sea, un ciclo completo de jugadas.

Ganador:

Como es de esperar, en un juego de varios jugadores, debe haber al menos un ganador (digo al menos porque hasta ahora no hemos tenido en cuenta el concepto de empate, pero podria existir). Dicho ganador, dependiendo de las reglas, puede estar determinado por diferentes factores, siempre luego de que el juego finaliza (ahi entra la condicion de ganado). Hemos hecho 2 implementaciones de como seleccionar un ganador:

MinValueWinner:

Esta implementacion selecciona al jugador con menor valor entre todas sus fichas para declararlo ganador. Dicho valor viene calculado con la implementacion de ficha que se este utilizando en la partida en cuestion.

MoreTokensWinner:

Esta implementacion selecciona al ganador como el jugador con mayor cantidad de fichas luego de que se acaba la partida, sin tomar en cuenta el valor de las mismas, solo cantidad en mano.