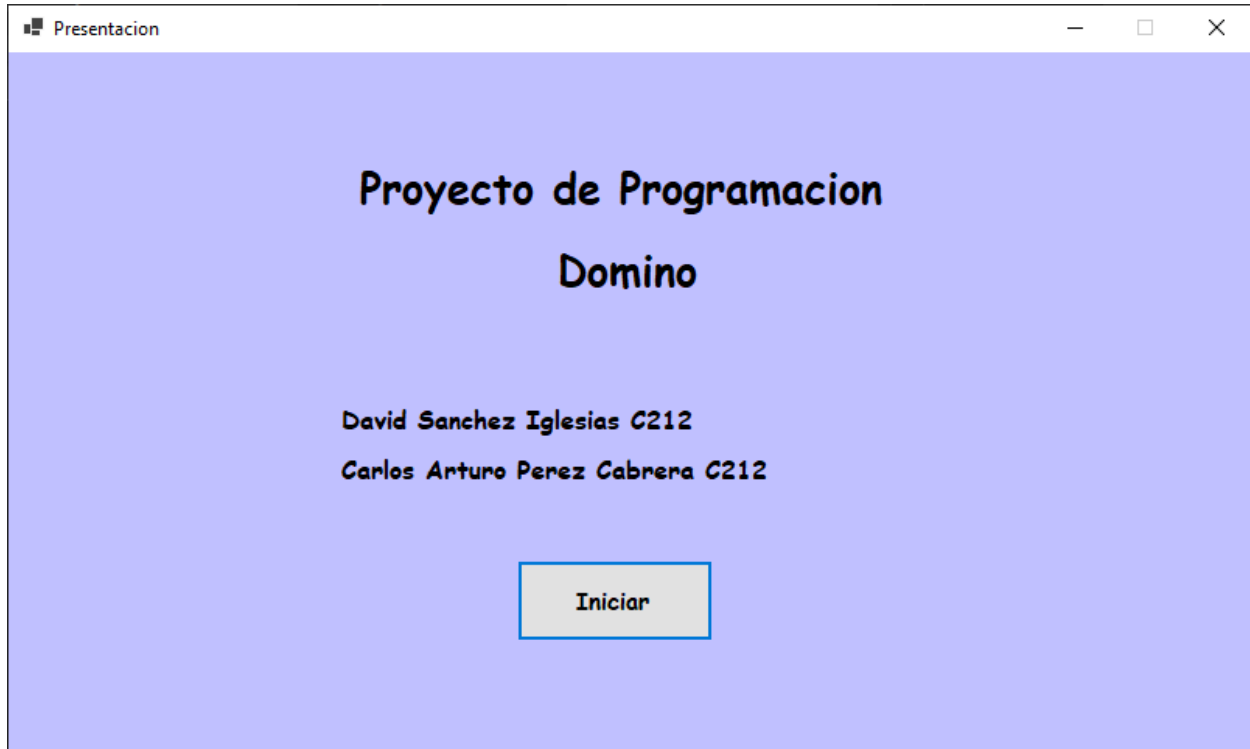


# PROYECTO DE PRO DOMINO

David Sánchez Iglesias C 212

Carlos Arturo Pérez Cabrera C 212

MATCOM



## Instrucciones:

Para usar esta aplicación y poder reproducir y observar el juego de dominó, deberá seguir unos sencillos y cortos pasos que aquí se explican.

Después de presionar el botón "Iniciar" de la pantalla de presentación, se abre la pantalla de configuración del juego. En esta nueva ventana, el usuario puede escoger y cambiar a voluntad los modos de juego, la condición para que finalice el juego, el orden en el que jugarán los jugadores, la forma en la que se repartirán las fichas a cada jugador y la forma de calcular la puntuación de cada jugador en dependencia de las fichas que haya jugado. En este proyecto se han implementado varias modalidades de cada una de las características modificables en esta ventana. Como modo de juego se han implementado los dominós con fichas hasta el doble 6 y hasta el doble 9.

**Opciones**

**Modo de Juego(por cantidad de fichas)**  
 FichasDe6  
 FichasDe6  
 FichasDe9

**Condicion de finalizacion**  
 FinalizacionPorPuntos\_50pu

**Orden de las jugadas**  
 OrdenNormal

**Forma Inicial de Repartir Fichas**  
 RepartoAleatorio

**Forma de Calcular la Puntuacion Final**  
 ContarFichasDoblesPor2

Siguiente

Como condición de finalización del juego, también fueron implementadas dos modalidades: “FinalizacionPorPuntos” establece que, si un jugador alcanza determinada cantidad de puntos (50 puntos) durante el juego, este termina y dicho jugador se establece como ganador; y “FinalizacionPorPase” establece que, si un jugador se pasa dos veces seguidas, se termina el juego y se determina el jugador con la mayor cantidad de puntos hasta el momento, como ganador. Es importante aclarar que un jugador obtiene puntos al jugar una ficha, teniendo en cuenta la manera en la que escogió que se calculara la puntuación de las fichas; en otras palabras, la puntuación se decide evaluando la ficha que se juega en cada jugada, y sumando el resultado a los puntos que el jugador haya acumulado hasta el momento. En “Orden de las jugadas” se han hecho también dos variantes, “OrdenNormal”, que mantiene el orden de turnos de los jugadores hasta que acabe el juego; y “OrdenCambiadoSiSePasa”, que establece que, si un jugador se pasa (no juega ninguna ficha) entonces el orden de turnos se invierte, ejemplo: juegan “jugador1”, “jugador2”, “jugador3” y “jugador4” en ese orden, siendo el jugador1 el primero en jugar; si cuando debe jugar el jugador3 este se pasa, el siguiente en jugar sería el jugador2, pues se invierte el orden de turnos. Si en el ejemplo anterior, después de que

el jugador3 se pase, el jugador2 también se pasa entonces el algoritmo vuelve a cambiar el orden de los turnos y busca el siguiente jugador que pueda jugar.

Respecto a la forma de repartir, se debe aclarar algo primeramente, y es que una cosa son las fichas que se le dan a los jugadores para que ellos escojan, y otra es la manera en la que estos últimos escogen las fichas de su “mazo”. La forma de repartir no determina qué fichas le tocan a cada jugador, sino a partir de qué conjunto de fichas los jugadores pueden escoger, y si ellos pueden ver o no los números de estas fichas.

Sabiendo esto, se programaron dos formas de repartir las fichas: aleatoriamente (equivale a darle a escoger a los jugadores de entre todas las fichas del juego, con las fichas boca abajo, o sea, el reparto usual), y el reparto boca arriba, que permite a los jugadores “ver” los números de las fichas y escogerlas a conveniencia. Por último, como forma de calcular la puntuación, se han creado tres criterios: las fichas dobles tendrán el doble de la suma de los números que posean, por ejemplo, el doble 6 valdría  $2 * 12$ , o sea, 24; otro criterio es contar el valor de las fichas dobles como negativo, esto es restar los puntos que poseen las fichas dobles del total de puntos que posee el jugador hasta el momento; y un tercer criterio sería sumar ambos números de cada ficha a la puntuación del jugador.

Luego, una vez configurado el juego, el usuario debe presionar “Siguiente” para acceder a la ventana de creación de jugadores. En dicha ventana se encuentran dos campos a rellenar: el primero: “Tipo de jugador”, determina la estrategia que empleará el jugador durante el juego una vez creado; el segundo: “Nombre del jugador”, proporciona un nombre a dicho jugador para diferenciarlo de los ya creados anteriormente o de los que se crearán luego, si ese es el propósito del usuario.

Respecto a los tipos de jugadores, en este proyecto se han implementado tres: el aleatorio, cuya estrategia es precisamente jugar una ficha aleatoria en cada jugada; el goloso, que siempre jugará la ficha con la que pueda obtener más puntos, dependiendo de la manera en la que se distribuyan los puntos; y el “inteligente”, que buscará la ficha que pueda jugar por un número y que el otro número sea lo más común posible dentro del resto de fichas que posee. Finalmente, después de escoger la estrategia del jugador y un nombre para él, se debe presionar “Agregar” para añadirlo a la lista de jugadores. El botón “Eliminar” se usa para borrar uno de los jugadores ya creados y

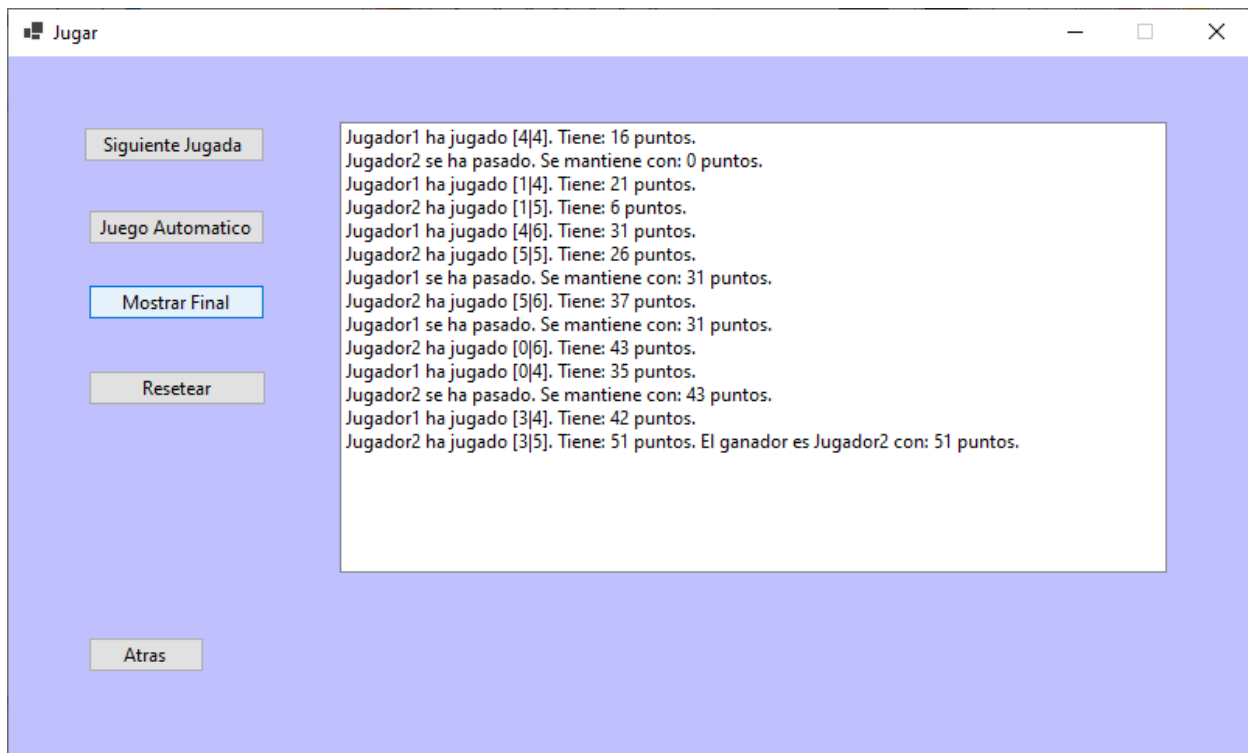
que se encuentre en la lista. Para borrar un jugador se debe dar click encima de un jugador en la lista de jugadores y luego presionar el botón “Eliminar”.

The screenshot shows a Java Swing window titled "Jugadores". The window has a light blue background. It contains two labels: "Tipo de Jugador" and "Nombre del Jugador". Under "Tipo de Jugador" is a dropdown menu with the following options: "JugadorGoloso", "JugadorAleatorio", "JugadorGoloso", and "JugadorInteligente". Below the dropdown is a list box containing "Jugador1 (JugadorAleatorio)" and "Jugador2 (JugadorGoloso)". Under "Nombre del Jugador" is a text field containing "Jugador2". To the right of these are four buttons: "Agregar", "Eliminar", "Siguiente", and "Atras".

Después de seleccionados los jugadores, el usuario ha de presionar el botón “Siguiente” con el fin de pasar a la ventana de juego.

En esta ventana se puede ver el juego y cómo se desarrolla: lo que juega cada jugador y los puntos que estos tienen en cada jugada.

Aquí hay 4 botones: “Siguiente Jugada” muestra la ficha jugada por el jugador al que le toca ahora, o sea, le da el turno al siguiente jugador en la cola; “Juego Automático” reproduce cada jugada de manera autónoma hasta que termine el juego o hasta que se presione el botón de nuevo; “Mostrar Final” pone en pantalla el juego completo y su resultado de manera instantánea; y “Resetear” reinicia el juego, manteniendo las condiciones y configuraciones establecidas en la ventana de configuraciones del juego, con la diferencia de que a los jugadores se les vuelve a repartir las fichas para que escojan.



#### Detalles técnicos:

Para el diseño del juego como tal, se crearon varias interfaces que definen cada elemento personalizable del juego. Cada configuración manipulable en la ventana de configuraciones del juego implementa internamente una interfaz. Así se tienen 5 interfaces: "IFicha" define la manera en la que se generarán todas las fichas del juego. Lo único que se debe definir a la hora de crear una nueva clase que implemente esa interfaz, es la cantidad de fichas que deberá tener cada jugador y el conjunto total de fichas que tendrá el juego; la manera en la que se generen dependerá del que la esté definiendo; "ICondicionDeFinalizacion" se compone de un solo método booleano que determina si el juego acabó o no, en dependencia de la condición que se haya elegido; "IOrdenDeLasJugadas" se compone de solo dos funciones: una que determina el jugador al que le toca jugar y un método "Reset" que reinicia el estado de la clase del orden que se haya escogido; "IFormaDeRepartir" conceptualiza la forma en la que se reparten las fichas a los jugadores; y "IFormaDeCalcularPuntuacion" define la manera en la que se determina el valor de las fichas.

Para el diseño de los jugadores, se usó una interfaz “IJugador”, que establece que todo jugador debe tener nombre, puntuación, un conjunto de fichas (con las que va a jugar), la manera de calcular la puntuación (definida como un delegado al que se le asigna la clase que implemente la interfaz “IFormaDeCalcularPuntuacion” que se haya escogido en la interfaz gráfica, en las configuraciones del juego), dos variables booleanas, una para saber si es su turno, otra para saber si ganó, un método para seleccionar las fichas al inicio del juego y un método para jugar, que devuelve la ficha con la que el jugador va a jugar.

Por otra parte, el juego como tal es un Enumerator de “Jugada” que recibe las instancias de cada clase, creadas durante la selección de las opciones de juego en la interfaz gráfica. Aquí “Jugada” es una clase destinada a imprimir en pantalla la jugada que se realizó, o sea, imprime el nombre del jugador, si se pasó, y si no, imprime la ficha jugada, si se trancó el juego, si alguien ganó y la cantidad de puntos del jugador que acaba de jugar. En el “MoveNext” del Enumerator es donde se determina cuál es el siguiente jugador que debe jugar, la ficha que dicho jugador va a jugar, si se finalizó el juego, si hubo ganador, y donde se imprime lo que sea que pasara en el turno; o sea, en el “MoveNext” se ejecuta todo lo que debe ocurrir en un turno.