

# Proyecto entrega 1

Julian Camilo Builes, Gabriela María Camacho y Abel Santiago Cortés

27 de abril de 2021

## Resumen

En este documento se presenta la interfaz creada por el grupo para el juego 2048.

## 1. Análisis del juego

En primer lugar es necesario describir el problema en términos y requisitos del cliente. Para esto, definimos que la programación por contrato en este algoritmo parte por tener una matriz de tamaño finito de  $4 \times 4$ . En dos de estas posiciones se almacenan aleatoriamente dos números que pueden tomar el valor de 2 ó 4. Los valores de esta matriz pertenecen al dominio de los números enteros, representados por potencias de 2.

Partiendo de estos datos, el algoritmo recibe cómo entrada los diferentes movimientos que puede generar el usuario en la matriz. Los movimientos pueden ser únicamente a la izquierda, derecha, arriba y abajo, al generar estos movimientos los números se desplazarán hasta el límite que exista en ese lado de la matriz. Al desplazar existen los siguientes escenarios:

- El número termina en el límite de la matriz si no hay más números de por medio.
- El número queda al lado de un número cuando el número adyacente no es el mismo.
- El número se agrupa con el número adyacente cuando el valor es el mismo, tomando el resultado en la misma celda.

Finalmente el algoritmo se compromete a notificar constantemente al usuario la sumatoria total que genere las agrupaciones a partir de los movimientos. El juego termina y la salida final es un valor booleano que indique cuando la matriz no permite más movimientos, lo que significa que ha perdido, ó que llegó a generar una agrupación entre dos valores, en donde su resultado es 2048.

Existen diferentes formas de implementar este algoritmo, para este primer acercamiento se propone un algoritmo de fuerza bruta. Para cada movimiento (izquierda, derecha, arriba y abajo) se llevará a cabo un recorrido en la matriz que evalúe los tres escenarios mencionados previamente. En cada uno de los movimientos se usarán los índices  $i$  y  $j$  para evaluar las condiciones y en caso tal, hacer el cálculo de las sumas. Además existirá un contador que irá sumando constantemente cuando se agrupen dos casillas de la matriz, informando al usuario del progreso en el juego.

## 2. Diseño de la interfaz

Para el desarrollo de la interfaz se utilizó Python con la extensión PyQt5 y es necesario tener instalado para la utilización del juego, para instalarlo abra su cmd y escriba el siguiente comando `pip install PyQt5`.

Se planteó una interfaz que permitiera jugar 2048, en este hay cuatro movimientos arriba, abajo, izquierda y derecha, cada uno está representado por un botón con su respectivo nombre. También, se tiene una matriz de tamaño  $4 \times 4$ , en donde se proyectan las "fichas/números" por último un se tiene la acumulación del puntaje que se lleva en una partida. Finalmente la interfaz se verá de la siguiente manera.

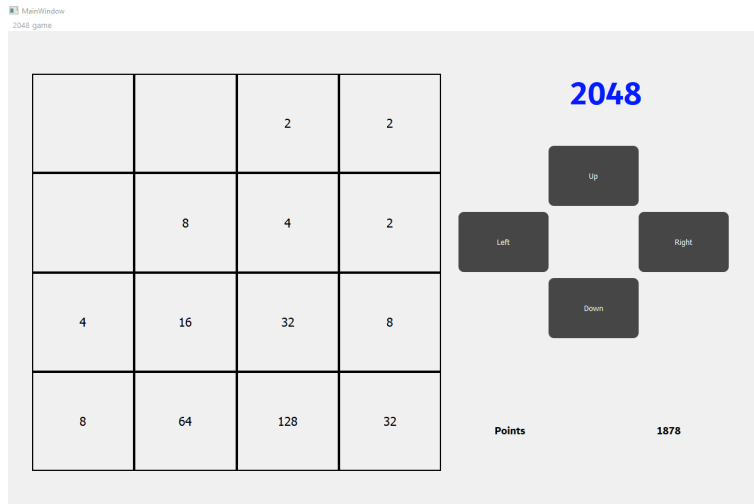


Figura 2.1: Interfaz

Para que la interfaz sea útil y se pueda jugar es necesario crear otra clase, en nuestro caso Game 2048, en esta se realiza todo el procedimiento, como se muestra en la figura 2.1.

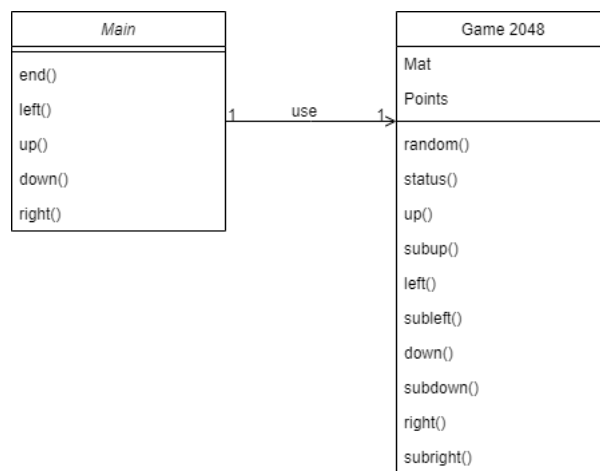


Figura 2.2: Diagrama de clases juego 2048

Como se puede observar la clase main, la cual es la interfaz gráfica que permite a los jugadores interactuar con el juego, tiene cuatros métodos los cuales corresponde al click recibido por pantalla sobre un botón específico, que indica su movimiento (up,down.left y right), por medio de este se comunica con la clase Game 2048, donde se realiza el procedimiento ilustrado en el siguiente diagrama de flujo.

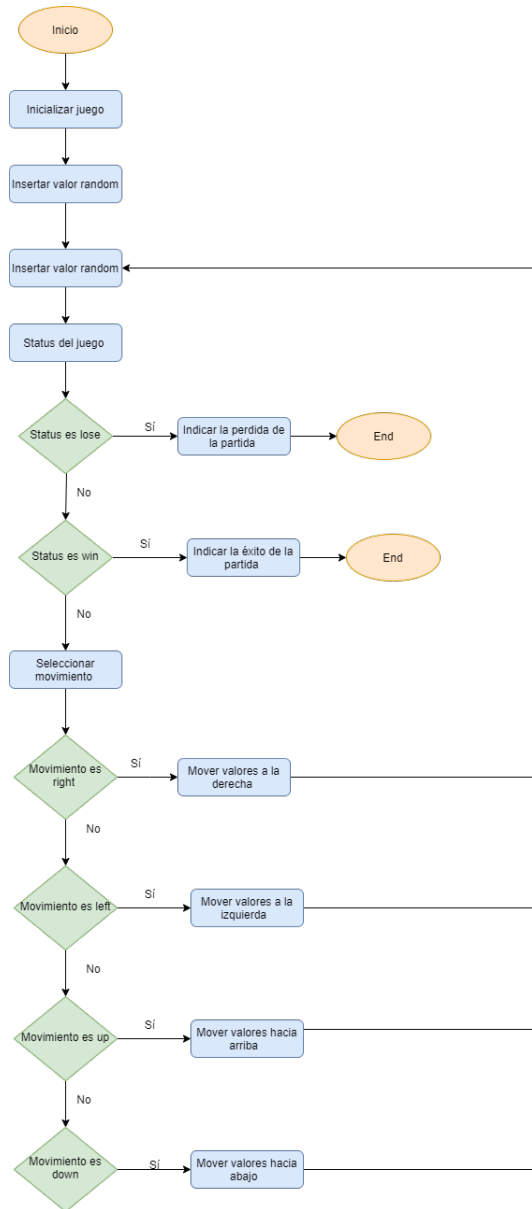


Figura 2.3: Diagrama de flujo juego 2048

Sin importar que movimiento se seleccione el código evalúa hacia donde debe mover las fichas, cuales debe sumar y de ser así acomoda nuevamente el tablero insertando la ficha nueva, por medio de las siguientes funciones(figura 2.2):

- Inicializar juego, se crea una matriz 4x4
- Insertar valor random, es decir en cualquier posición se inserta un 2 o un 4.
- Status del juego, obtiene el estado del juego es decir que si se perdió, se gano o existe la posibilidad de hacer más movimientos, es decir continuar la partida.
  - Sí, el estado es lose, se indica la perdida de la partida y se termina el juego.
  - Sí, el estado es win, quiere decir que el jugador 2048 se indica que gano la partida y se termina el juego.
- En caso de que no sea ninguno de los casos anteriores, es decir continúe, se le da la opción al jugador de elegir el movimiento.
  - En caso de que sea right, se moverán todos los valores a la derecha.
  - En caso de que sea left, se moverán todos los valores a la izquierda.
  - En caso de que sea up, se moverán todos los valores hacia arriba.
  - En caso de que sea down, se moverán todos los valores hacia abajo.
- Finalmente se agrega un numero aleatorio en cualquier posición por medio de la función random y se realiza nuevamente el proceso.