Documentación técnica

**Flujo de ejecución del programa:**

Al arrancar el juego lo primero que hace es declarar y setear las variables que se usarán a lo largo del programa, así como iniciar la instancia de ventana para mostrar el menú.

Luego entra en un loop para verificar las entradas del usuario y determinar la opción elegida, el cual verifica según la ocurrencia de un evento keydown el presionado de una tecla y luego verifica cual fue esa tecla para determinar que hacer a continuación.

Acorde a la selección del usuario el juego puede enviarlo a una partida, una tabla de puntajes, un tutorial del juego o salir del programa.

Sección Partida:

La sección partida (start) opera con un flujo similar al del menú: luego de setear las variables necesarias para la operación de la sección, entra en un bucle que inicializa los objetos y variables necesarios para una partida y actúa como “precursor” del bucle interior, el cual corresponde a cada partida de manera individual.

Cuando el bucle interior interrumpa su ejecución, es decir, la partida termine, ya sea saliendo con la tecla escape, por tiempo o porque algún jugador ganó, el bucle precursor retomará el control y acorde al tipo de salida del bucle de partida reinicializará los objetos y variables o volverá al menú.

Sección de Puntajes:

La sección de puntajes (score) es la encargada de mostrar los mejores puntajes de las partidas que se hayan jugado. Opera cargando 10 puntajes almacenados en un archivo, mostrándolos en pantalla y aguardando a que el usuario presione la tecla escape para volver al menú.

Sección Tutorial:

La sección tutorial (how to game) se encarga de mostrarle al usuario los controles del juego, el flujo de esta sección es simple, una vez que se renderiza en pantalla el texto de los controles, el programa queda a la espera de que el usuario presione la tecla para volver al menú.

**Problemas encontrados:**

Angulo de rebote para la pelota:

Uno de los primeros problemas con los cuales nos encontramos durante el desarrollo del juego fue determinar el ángulo y velocidad de rebote de la pelota cada vez que esta chocaba contra una pared o un jugador. La resolución a este problema lo encontramos en la librería pygame misma, la cual trae integrada una función para encargarse de las colisiones.

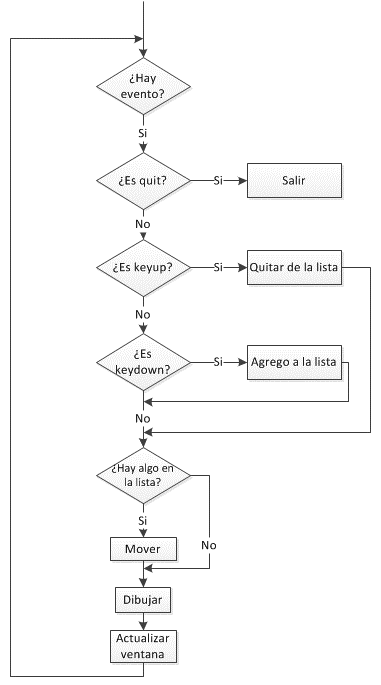
Una vez resuelto ese problema quedo la cuestión de como manejar el rebote, es decir, dependiendo de donde pegase hacia donde iba a salir. Esto se resolvió partiendo el “rectángulo” de los sprites de los jugadores en 4, si la pelota pega en la sección superior del rectángulo rebota hacia arriba, sino hacia abajo, y si rebota en la sección delantera rebota hacia adelante, de lo contrario hacia atrás.

Las colisiones con las paredes se manejan de manera similar, una vez que se detecta una colisión se revisa si la pelota golpeó el techo o el piso del rectángulo que conforma al área de juego y acorde a eso rebota hacia arriba o abajo, los lados verticales siempre conservan el movimiento en el eje Y, aunque lo invierten en el eje X.

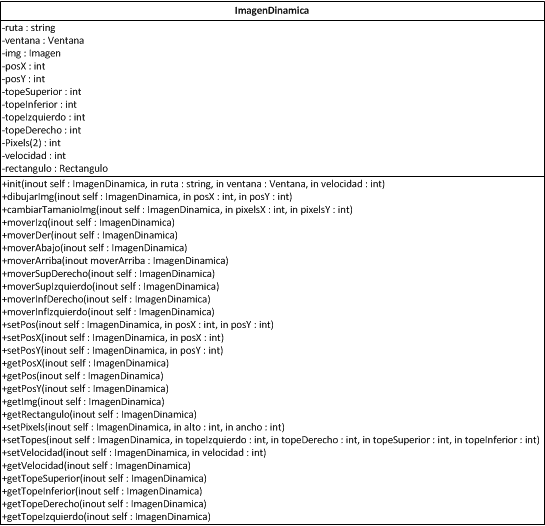
Presionado de múltiples teclas al mismo tiempo:

Otro problema encontrado durante el desarrollo fue manejar el presionado de varias teclas al mismo tiempo, lo cual es vital para que pueda haber más de un jugador manejando su personaje al mismo tiempo y para que el movimiento de estos en diagonal sea posible.

Esto se resolvió implementando una lista de teclas presionadas, la cual se actualiza a medida que van llegando eventos de presionado o levantado de las teclas que corresponden al movimiento de cada jugador. Esta lista luego se revisa para determinar, por jugador, las teclas presionadas y determinar la dirección en la cual se moverá cada uno.



**Diagrama de clases:**

Por su estructura, el programa no requirió más que una clase, la cual esta representada a continuación:

Esta clase se llama “ImagenDinamica” porque me sirve para usarla en todas las imágenes del juego que se van a mover, es por eso que se llama dinámica ya que para las imágenes que están “estáticas” con la librería de pygame ya es suficiente. En vez de hacer una clase jugador y una clase pelota, integramos esas 2 clases en una sola ya que el jugador y la pelota se mueven y tienen características parecidas, ellos formaran parte de la clase ImagenDinamica, esta clase contiene topes izquierdo, derecho, superior e inferior a los cuales los objetos de esta clase se pueden mover, el jugador 1 tendra un tope, el jugador 2 otro tope y la pelota tendrá otro tope. Tambien tiene atributos como velocidad, pixels de la imagen, posición actual de la imagen, métodos para poder moverse, para cambiar tamaño y dibujarla en pantalla.