

Instituto Tecnológico de Costa Rica

Taller de Programación

II Tarea Programada

Profesor:

Jeff Schmidt

Estudiantes:

Andrés Quirós

Isaac Araya

Grupo 1

2018

Índice

Introducción	3
Descripción del problema	4
Diagrama de clases	5
Análisis de resultados	6
Dificultades encontradas	12
Bitácora de actividades	13
Estadística de tiempos	18
Conclusiones	19

Introducción

En esta segunda tarea programada se utilizará el lenguaje de programación Python para crear un programa que simule el videojuego Pong mediante ciertos métodos y programas para hacer que funcione de la manera indicada. Para poder realizar dicha tarea, se investigará profundamente sobre las diferentes áreas que esta posee y sobre las maneras que hay para realizarlas, buscando la más eficiente para poder utilizarla. Para la interfaz gráfica se va a utilizar el módulo Pygame, el cual facilita de gran manera el diseño de videojuegos en comparación con otras herramientas como Tkinter. Además, se deberá investigar e implementar matrices para utilizarlas como el tablero del juego, el cual es donde las paletas y la bola se moverán. El uso de Git y Github son de gran importancia para este trabajo, por lo que también se utilizarán para mantener un respaldo local y remoto del trabajo realizado día a día.

Este trabajo es de mucha relevancia ya que nos ayudará a seguir desarrollando nuestras habilidades de programación y aumentar nuestro conocimiento sobre nuevos programas y herramientas que nos servirán en un futuro con otros proyectos tanto personales como profesionales. Será de vital importancia realizar esta tarea de la manera más comprensible y completa posible ya que será una parte crucial para la tercera tarea programada, por lo que vamos a tener esto en cuenta a la hora de realizar este trabajo.

Descripción del problema

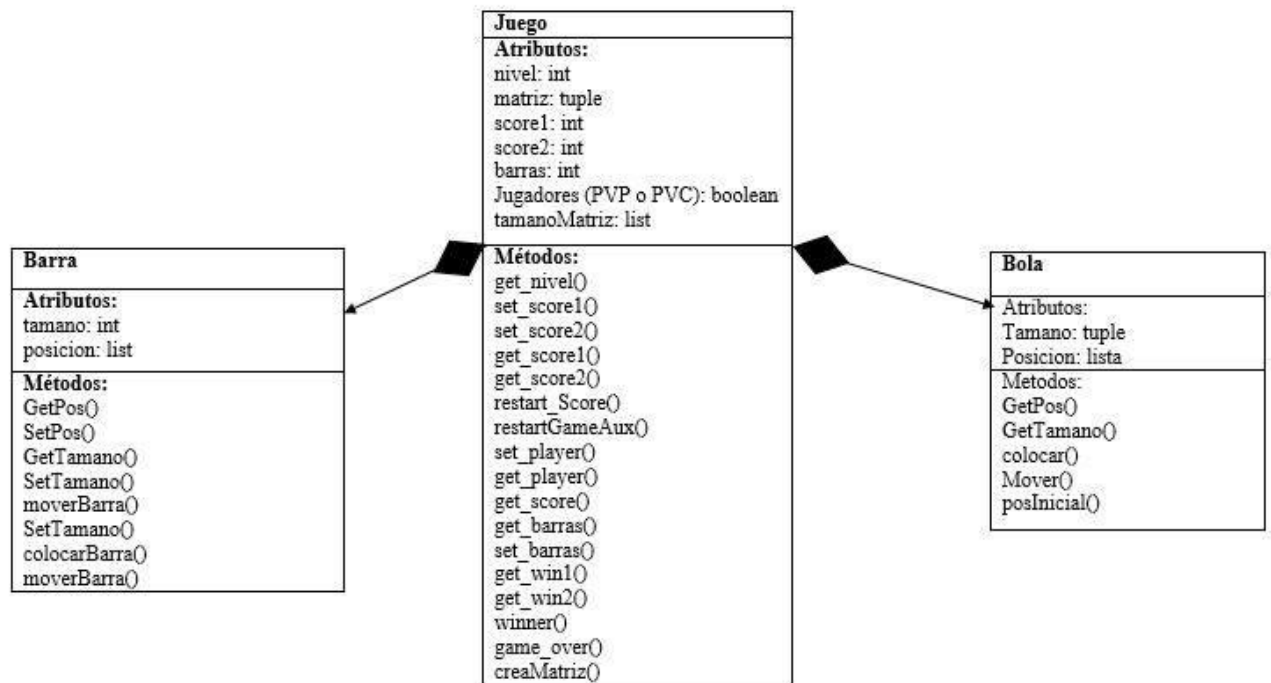
Se va a realizar un programa que simule el famoso videojuego Pong, el cual consta en una partida al estilo ping pong, donde un jugador tiene una paleta y debe derrotar a su oponente obteniendo la cantidad de puntos necesarios para ganar, esto se logra haciendo que tu rival no logre devolver la pelota y que esta siga recto hasta el límite, aquí es donde se gana un punto.

En el menú principal del juego van a haber 2 opciones a elegir, jugar contra la computadora o jugar contra otra persona, al elegir van a aparecer otras opciones que van a definir si se va a jugar con una(single) o dos paletas (doubles). Al iniciar el juego, va a aparecer el tablero donde van a estar las paletas, la pelota, el tiempo y el puntaje de cada jugador; este tablero es una matriz de 25x40 en donde van a interactuar todos los objetos y la interfaz es la encargada de reflejar todas estas acciones en la pantalla. Las paletas van a tener un tamaño de 9, en donde las primeras 3 van a golpear la bola en diagonal hacia arriba, las 3 del medio la van a golpear en línea recta y las 3 últimas la van a mover en diagonal hacia abajo. En el case de los “doubles”, van a haber 2 paletas de cada lado que se moverán al mismo tiempo y que se manejarán con los mismos botones en el teclado.

En el modo PVC (contra la computadora) el juego va a constar de 3 niveles, en los cuales mientras más se va avanzando más rápido se va a mover la pelota, más difícil va a ser jugar contra el computador y las paletas se van a reducir más. En el nivel uno la paleta va a tener un tamaño de 9, en el dos de 6, y en el tercero de 3. Si el jugador gana los 3 niveles ganará la partida y en el caso de que pierda alguno deberá comenzar desde de nuevo en el nivel 1.

En el modo PVP (contra otro jugador) el ganador va a ser el que gane 2 partidas de 3, aquí también aplica lo de la disminución del tamaño de las paletas y el aumento de la velocidad de la pelota conforme se avance en las partidas, por lo que, si hubiera un empate y se tuvieran que ir a una tercera partida, la bola irá lo más rápido posible y las paletas tendrán un tamaño de 3.

Diagrama de clases



Análisis de resultados

Menú principal del juego:



El programa al iniciarse desplegará una ventana la cuál será el menú principal del juego, en donde se pueden escoger diferentes modos con diferentes opciones según desee el jugador. Para la implementación de esta ventana, se utilizó una imagen creada en Paint en donde se dibujaron los objetos y el título observados en esta, y al colocarla en Pygame solamente se le colocaron los 2 botones que se pueden observar dentro de la bola.

Modos de juego:



En el menú principal hay 2 botones los cuales establecen los modos de juego disponibles. Aquí el jugador puede elegir entre jugar contra la máquina o jugar contra otro jugador. En cualquiera de los dos casos, al apretar el botón del modo que se desea jugar, aparecerán otras 2 opciones que corresponden a que si se

quiere jugar las cuales son con 1 barra o con 2 barras, y la opción escogida aplicará para ambos jugadores.

Definición de ganador:

Cada uno de los modos están definidos en la función Mainloop() en la cual hay 3 while's que definen los 3 niveles del juego y dentro de cada while hay if's que definen si se está jugando con una o dos paletas. En cada uno están establecidos los casos en que se ganan o se pierden una partida

-Contra la máquina: El jugador tendrá que jugar 3 partidas contra la máquina, cada vez que gane una subirá de nivel. Entre más avanzado el nivel, mayor será la velocidad de la bola y menor será el tamaño de las barras. Si el jugador pierde alguna de las partidas habrá un Game Over y deberá empezar de nuevo desde el nivel 1. Si el jugador logra ganarle a la máquina en los 3 niveles, acabará el juego y tendrá la opción de iniciar de nuevo o de salirse del juego

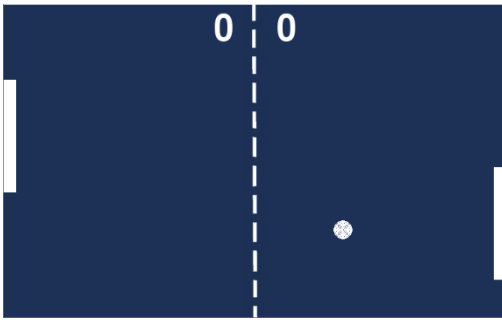
-Contra un jugador: Los jugadores deberán enfrentarse en máximo 3 partidas para determinar quién gana el juego. Este modo también posee los 3 niveles en los cuales entre más alto sea, mayor será la velocidad de la bola y menor será el tamaño de las barras. En ganador se va a decidir según quien sea el primero en ganar 2 partidas de las 3 posibles.

Nivel 1



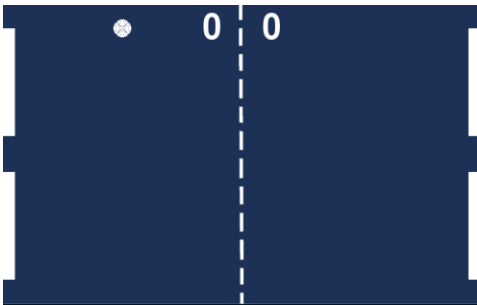
Al empezar cualquier modo de juego aparecerá esta pantalla que indica que el nivel uno está a punto de iniciar. Luego iniciará el juego, el cual será diferente según el modo escogido y la cantidad de paletas:

-1 paleta:



En ambos modos, cuando se juega solo con una paleta, esta tendrá un tamaño de 9 (la cual sería de aproximadamente 180 píxeles en la pantalla) por lo que será relativamente fácil jugar en este nivel. Su posición inicial en la matriz será la de Pos ([8,0]) para la paleta de la izquierda y de Pos ([8,39]) para la paleta de la izquierda. Además, la velocidad de la bola será de 0.05 por lo que será fácil poder seguirla. Todo esto está establecido en la función Mainloop() del código del juego

-2 paletas:



En ambos modos, igual que con una paleta, ambas paletas para los 2 jugadores tendrán el mismo tamaño de 9 (180 píxeles). Para el jugador del lado izquierdo, las posiciones iniciales de sus paletas serán de Pos1 ([2,0]) y Pos2 ([14,0]). Y para el jugador del lado derecho, las posiciones iniciales de sus paletas son de Pos1 ([2,39]) y Pos2 ([14,39])

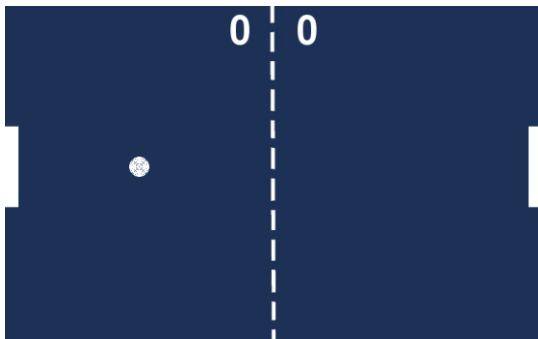
Al terminar el nivel y definir el ganador de la partida, el conteo de los puntos se reiniciará y se pondrá en cero, y al mismo tiempo se pasará al nivel 2 del juego

Nivel 2:



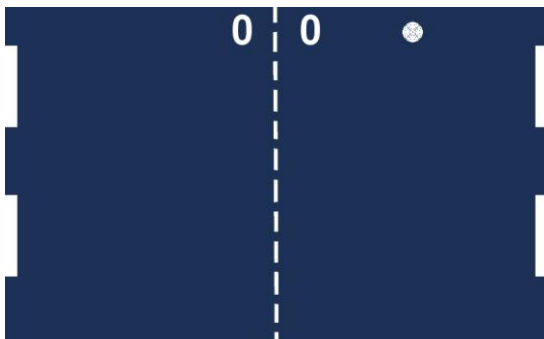
Luego de finalizar el nivel 1, aparecerá esta pantalla que indica que el nivel 2 está a punto de comenzar, el cuál sufrirá ciertos cambios con respecto al nivel 1 que afectará a la jugabilidad del juego durante este nivel, para así poder aumentar la dificultad de este. Se podría decir que esta sería la dificultad media.

-1 paleta:



Con respecto al nivel uno, el nivel 2 sufrirá ciertos cambios. La paleta de cada jugador sufrirá un cambio de tamaño, la cual pasará a disminuir de 9 (180 pixeles) a 6 (120 pixeles) por lo que se tendrá que ser más rápido al mover la barra ya que cubrirá menos espacio. Además, la bola aumentará la velocidad de 0.05 a 0.055 por lo que será más difícil seguirle el ritmo. La posición inicial de la paleta izquierda es de Pos ([9,0]) y el de la izquierda será de Pos ([9,39])

-2 paletas:



Igual que con una paleta, el tamaño de las cambiará ya que disminuirán de 9(180 pixeles) a 6 (120 pixeles), por lo que será un poco más difícil porque cubrirán

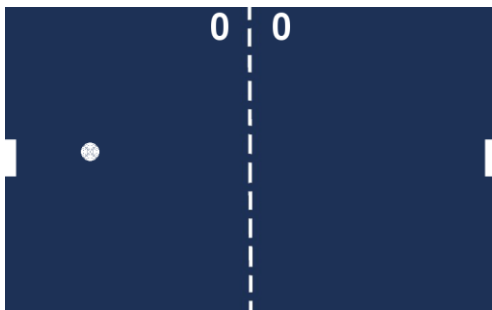
menos espacio, pero más fácil que con una paleta porque aquí son 2. La bola, igual que con 1 paleta, pasará a aumentar su velocidad de 0.5 a 0.055. Las posiciones iniciales de las paletas del lado izquierdo serán Pos1([3,0]) y Pos2([14,0]), y las del lado derecho serán de Pos1([3,39]) y Pos2([14,39]), por lo que habrá una separación de 9 espacios entre las dos paletas de cada lado.

Nivel 3:



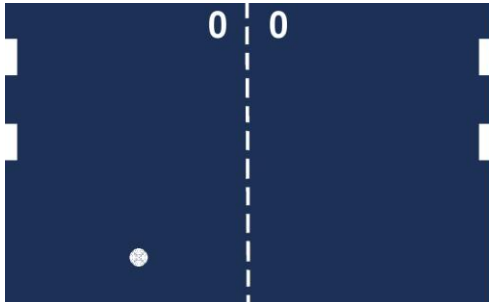
Al terminar el nivel 2 del juego, aparecerá esta pantalla la cual indica que el nivel 3 está a punto de comenzar. Este nivel sufre varios cambios con respecto a los anteriores con respecto a la jugabilidad del juego.

-1 paleta:



El nivel 3 posee ciertas diferencias que los hacen mucho más difícil que los niveles anteriores, ya que los tamaños de las paletas vuelven a sufrir un cambio y se vuelven aún más pequeñas, pasando de tener un tamaño de 6(120 pixeles) a 3(60 pixeles), por lo que será más difícil pegar la bola en el lugar correcto. Además, la velocidad de la bola vuelve a aumentar, ahora va de 0.055 a 04, por lo que será más difícil seguirle el paso. La posición inicial de la paleta de la izquierda es Pos ([11,0]) y el de la derecha es de Pos ([11,39]).

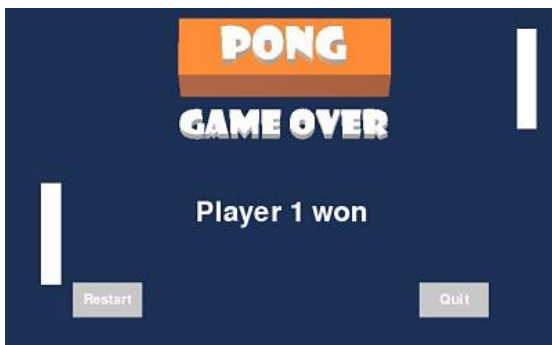
-2 paletas:



Al igual que con una paleta, las paletas de los jugadores tendrán una disminución de tamaño, la cual será de 6 (120 pixeles) a 3 (60 pixeles), por lo que ahora será aún más difícil controlarlas. Además, la bola sufre el mismo cambio de velocidad, el cual va de 0.055 a 0.4. Las posiciones iniciales de las paletas de la izquierda son de Pos1([3,0]) y Pos2([14,0]), y las del lado derecho tienen de posición inicial Pos1([3,39]) y Pos2([14,39]), por lo que también poseen una diferencia de 9 espacios de la matriz entre ellas.

Game over:

Al terminar el juego, se definirá el ganador y aparecerá esta pantalla, donde el juego puede decidir entre volver al menú principal o salir de juego



Dificultades encontradas

El primer problema que tuvimos fue con el programa Git ya que los cambios que hacíamos en los archivos no se guardaban en el respaldo remoto que estaba en Github, por lo que no podíamos seguir avanzando hasta solucionarlo.

Investigamos en internet e hicimos varias pruebas y después de unas horas logramos resolverlo. El problema es que no agregábamos los archivos antes de hacer cada commit.

Otro inconveniente, fue que no se lograba imprimir la matriz correctamente en la consola de Python. Luego de un rato de hacer pruebas se logró que la matriz se pudiera mostrar correctamente, además, se pudo colocar la bola y las barras como variables en la matriz para que se pudieran diferenciar del espacio vacío y así identificar su posición. El segundo problema que ocurrió relacionado a esto, fue que se podían mover los elementos, pero solo metiendo valores específicos para la localización. Se logró solucionar al establecer una función que permitiera sumar valores a las barras para que se movieran hacia abajo o arriba, pero la bola se dejó para después para poderlo hacer junto a la interfaz.

Bitácora de actividades

Sábado 5 de mayo

-Análisis de requerimientos: se empezó con la búsqueda de información de diferentes aspectos de este trabajo, como la interfaz, la matriz y el uso de objetos. Principalmente, se vio videos explicativos y código para la utilización de Pygame como la interfaz gráfica escogida a utilizar en esta tarea programada.

Domingo 6 de mayo

-Análisis de requerimientos: se empezó a utilizar Git para ver cómo se podía utilizar en este trabajo, hubo problemas a la hora de utilizarlo ya que al hacer cambios en los archivos que habían, estos no se guardaban en el respaldo remoto que estaba en Github; a pesar que se trató de solucionar, no se pudo resolver por completo.

Lunes 7 de mayo

-Análisis de requerimientos: luego de un rato, se logró solucionar el problema que había con Git y ahora los archivos sí guardaban sus cambios en el respaldo remoto de manera inmediata al usar el "push". Con este problema solucionado, se logró establecer un sistema de guardado tanto local como remoto para que ambos integrantes pudieran utilizar estos archivos sin ningún inconveniente.

-Elaboración de documento: se empezó con la elaboración de los diagramas de clases, los cuales son parte de la documentación externa. Se utilizó la página sketchboard.me, la cual permite diseñar de manera rápida y sencilla la elaboración de estos diagramas y que, además, permite que varios usuarios hagan modificaciones al mismo tiempo, por lo que ambos pudimos hacer los diagramas al mismo tiempo.

Martes 8 de mayo

-Diseño de la aplicación: se empezó con el diseño y elaboración del menú principal del juego. Utilizando el módulo Pygame, se diseñó este menú el cual contiene el título del juego, un diseño del juego de fondo y los botones de PVC y PVP, para poder elegir el modo de juego a ejecutar.

-Programación: utilizando como base los diagramas de clases, se empezó a programar las clases correspondientes a los objetos que se van a utilizar en este trabajo, en específico se definieron los atributos y métodos de las clases barra,

bola y juego. A algunos métodos, especialmente los del juego, no se les pudo definir nada ya que estos se van a ir definiendo conforme se vaya avanzando en el desarrollo del programa.

Miércoles 9 de mayo

-Programación: se hicieron varios cambios y se agregaron varias líneas de código a las clases ya establecidas. Estos cambios corresponden a la definición de ciertos métodos, especialmente de la clase juego. Todavía hay métodos que no se pueden definir por lo que se dejaron en blanco. Los principales métodos con cambios fueron: timer, winner, game_over y score.

-Elaboración de documento: se inició con la escritura de la documentación externa del proyecto. Debido a la etapa en la que se encuentra el proyecto, solo se definió la introducción, la descripción del problema, las dificultades encontradas hasta ahora, la bitácora de los días hasta ahora trabajados y la estadística de tiempo con las horas hasta ahora invertidas en este trabajo

Jueves 10 de mayo

-Programación: se hicieron cambios y añadidos mínimos en las clases, especialmente en la del juego y se intentó definir la matriz para poder mostrarla en la consola.

Sábado 12 de mayo

-Investigación de funciones: se investigó sobre la aplicación de matrices en Python y Pygame para poder definirla en el programa. Además, se buscó sobre el movimiento y colocación de objetos para esta matriz.

Domingo 13 de mayo

-Programación: Se realizó un avance importante en el establecimiento de la clase juego y sus métodos, en donde se define el ganador de la partida y cuando ocurre un game over en cualquier modo. Además, se hicieron algunos cambios mínimos en otras funciones que están en diferentes clases.

-Diseño de aplicaciones: se inició la búsqueda de sonidos para aplicarlos en el juego, como por ejemplo el de la bola cuando golpea la barra, cuando golpea una pared, al obtener un punto, al ganar y al perder una partida. Además, se buscaron canciones para poner en el menú principal del juego. Se disminuyó el tamaño de

estos audios para que no gasten mucho espacio en los 2 megabytes disponibles para el trabajo.

Lunes 14 de mayo

-Programación: Se comenzó con la creación de la matriz en el juego y se logró que se mostrara en el Shell de Python para poder observar sus cambios. Además, se hicieron cambios y se definieron métodos en la clase bola y en la de barra, en específico, en la clase bola se logró que se mostrara en la matriz y que se moviera con valores específicas en esta. Se definieron las funciones de colocación y movimiento para esta clase. Para la matriz, se definió que una función que imprima la matriz y otra que la actualiza al mover algún objeto.

-Elaboración de documento: se continuó con la elaboración del documento, se actualizó la bitácora de trabajo y la tabla de horas laboradas. Además, se terminó de redactar la introducción

Martes 15 de mayo:

Programación: Se continuó con la programación de los métodos de las clases. Se logró establecer un movimiento para la barra, pero este no era continuo ya que había que apretar la tecla varias veces para poderlo mover. Además, se logró vincular el movimiento de las barras con teclas específicas en el teclado de la computadora. Se empezó a definir el movimiento de la bola, pero no se pudo avanzar mucho ya que se tenía que lograr que esta se moviera continuamente, que se moviera en diferentes direcciones y que cambiara su dirección al golpear una pared o a una barra.

Diseño de la aplicación: Se colocaron todos los sonidos correspondientes a la mayoría de acciones a las cuales se les podían aplicar en el juego. En específico, se colocó sonido al golpe de la bola con las barras de los jugadores, al golpe de la bola con la pared superior e inferior del tablero, al ganarle un punto al oponente, al cambio de nivel en una partida y la música del menú principal del juego.

Miércoles 16 de mayo

-Programación: Con respecto a este aspecto del proyecto, se definieron los niveles del juego tanto en el modo de 1 jugador como en el de 2 jugadores. Se definieron dentro de la función mainloop() 3 whiles que definen cada nivel del juego en cualquiera de los 2 modos, en cada uno se define la longitud de las paletas, la velocidad de la bola, la cantidad de puntos para pasar al siguiente nivel y cual jugador ganó la partida, lo cual definirá si se seguirá jugando o si es game over,;

además, de que en cada una está definida el movimiento de las barras con las teclas de la computadora ya establecidas. En el caso del nivel 1, se definió que las barras tuvieran un tamaño de 9(180 pixeles) y que la bola tuviera una velocidad de 0.05 según lo definido en su respectiva clase; luego en el nivel 2 se estableció que las barras tuvieran un tamaño de 6(120 pixeles) y que la bola tuviera una velocidad de 0.055; y por último en el nivel 3 las barras tienen un tamaño de 3(60 pixeles) y la bola se mueve con una velocidad de 0.04.

-Pruebas: Se hicieron diversas pruebas con respecto a cómo funcionaba el juego y el código que se tenía hasta el momento para tener una vista general de lo que se tenía hecho y que era lo que faltaba de hacer o de arreglar del programa para que pudiera funcionar de la mejor manera posible, además de ver que arreglos se podían hacer para adaptar las nuevas funciones que se iban a agregar.

Jueves 17 de mayo

-Programación: Se continuó estableciendo funciones para el funcionamiento del juego, específicamente, se terminó de definir la función que permite que dependiendo de donde pegue la bola en la barra, así sea la dirección que la bola va a agarrar. Para esto, se tuvo que definir en cada barra por separado en la función colisión, ya que cada una tiene una posición diferente por lo que la dirección podría variar. En la función se definió un vector "x" y uno "y" que definen si se tienen que invertir los valores de sus ejes o no para el movimiento de la bola a la hora de tocar la barra. Se dividió la barra en 3 partes, y dependiendo de cuál parte tocara la bola, así serían los signos de los vectores del movimiento de la bola y determinarían como se debería mover.

-Elaboración de documento: Se siguió actualizando el documento respectivo con la documentación externa del trabajo. Se siguió con la bitácora de trabajo, detallando que se hizo y en que se desarrolló en el programa en cada día trabajado, además, también se actualizó la tabla de horas reales trabajadas en el proyecto en los días respectivos.

Viernes 18 de mayo

-Investigación de funciones: Se investigó sobre la implementación de una posible "inteligencia artificial" de la bola para poder utilizarla en el modo de un jugador, la cual en realidad sería que la barra siguiera la dirección de la bola y que de manera aleatoria fallara en ciertas ocasiones para no hacer a la CPU completamente imposible de vencer. Se buscaron diferentes ejemplos de códigos para ver cómo

se empezaba con el código de esta función y de cómo aplicarla a la matriz, pero no hubo mucho éxito.

Sábado 19 de mayo

-Programación: Se inició con la programación de la lógica del movimiento que va a obtener la dirección en que se va a mover la barra de la CPU en el modo de un jugador. Se empezó a definir la función con un par de if's que establecen la cantidad de barras en la partida y dentro de cada uno hay un par de while's que se establecen según la posición de la barra y la de la bola. No se logró completar ya que daba errores a la hora de iniciar el modo de juego, el cual daba un error de posición en las barras, y en otras ocasiones la consola de Python se quedaba pegada al intentar mover cualquier barra.

Domingo 20 de mayo

-Programación: Con respecto a este aspecto, se corrigieron errores en varias funciones del código, especialmente con el movimiento de las barras y de la bola en el modo de juego de un jugador. Se terminó de hacer la función que hacía que la barra de la CPU siguiera a la bola y que pudiera fallar con cierta aleatoriedad para no hacerla imposible de vencer. Se terminó por completo de hacer el código de todo el juego, por lo que al finalizar el juego funcionaba perfectamente cómo debía hacerlo. Luego se procedió a revisar todo el código para ver que se podía cambiar o quitar para que quedará lo más ordenado posible.

-Elaboración de documento: Se continuó con la elaboración del documento al cual se le actualizó el diagrama de clases, ya que estas cambiaron mucho desde que se creó la primera versión, además se hizo todo el análisis de resultados que explica cómo funciona todo el programa y se añadieron las conclusiones de ambos estudiantes, también se terminó de redactar la bitácora de actividades y la tabla de horas trabajadas se actualizó.

-Documentación interna: Se completó la documentación de todas las clases y funciones del código que todavía no habían sido explicadas completamente y se añadieron otros comentarios a las que no tenían nada para poder explicarlas bien.

-Pruebas: Se hicieron todas las pruebas del juego posibles, para poder probar cada uno de los 3 niveles y su funcionamiento dentro del juego, además de probar cada una de las paletas, tanto simples como dobles, en los dos modos de juego para asegurarse de que no generaran ningún problema, así mismo la bola y su movimiento dentro del tablero.

Estadística de tiempo

Función	Andrés Quirós	Isaac Araya	Total
Análisis de requerimientos	5	3	8
Diseño de la aplicación	4	3	7
Investigación de funciones	4	4	8
Programación	20	28	48
Documentación interna	1	3	4
Pruebas	2	2	4
Elaboración del documento	9	4	13
TOTAL	45	47	92

Conclusiones

Isaac Araya:

En esta tarea programada se nos complicaron muchas cosas que al final lograron enseñarme bastante. La definición de un modelo de objetos no me quedó muy clara durante las clases que lo vimos, pero por medio de esta tarea logré comprender de buena manera la programación de objetos. De igual forma esta tarea me ayudó a mejorar mi lógica de programación. A todo esto, se le puede sumar la utilización de la plataforma GitHub. Esto nos presentó un reto debido a que trabajar en equipo suele ser complicado, pero es muy importante aprender a hacerlo con ojos hacia nuestro futuro y considero que este proyecto nos ayudó a mejorar el trabajo cooperativo y conjunto con el mismo objetivo además de aprender a distribuir funciones y dividir las tareas.

Andrés Quirós

Durante este trabajo se pudieron aprender y fortalecer gran cantidad de actitudes y cualidades que no solo nos beneficia en el ámbito personal, sino también son de gran ayuda tanto en el área social como laboral. Los valores como el trabajo en equipo, respeto, responsabilidad, tolerancia entre otros fueron demostrados durante este tiempo, ya que había que fortalecer el trabajo en equipo para poder hacer correctamente este proyecto, y aunque a veces puede ser difícil trabajar con otras personas, no enseña que muchas veces es necesario y por esto mismo se nos da a convivir con estas condiciones. Con respecto al proyecto en sí, fue muy interesante y entretenido, aparte de todo el estrés, realizar un videojuego desde cero por primera vez en la vida, por lo que lo aprendido durante esta segunda tarea programada tiene mucho valor personal y como estudiante, ya que se aprendió a utilizar nuevos programas y a utilizar diferentes lógicas para tener varios puntos de vista a la hora de programar-