

Instituto Tecnológico de Costa Rica	Segundo Proyecto Programado
Escuela de Ingeniería en Computación	Pong
CE-1102 Taller de Programación	Prof: Jeff Schmidt Peralta
I Semestre 2018 4 de mayo de 2018	Consultas: grupo facebook correo: jschmidtcr@gmail.com

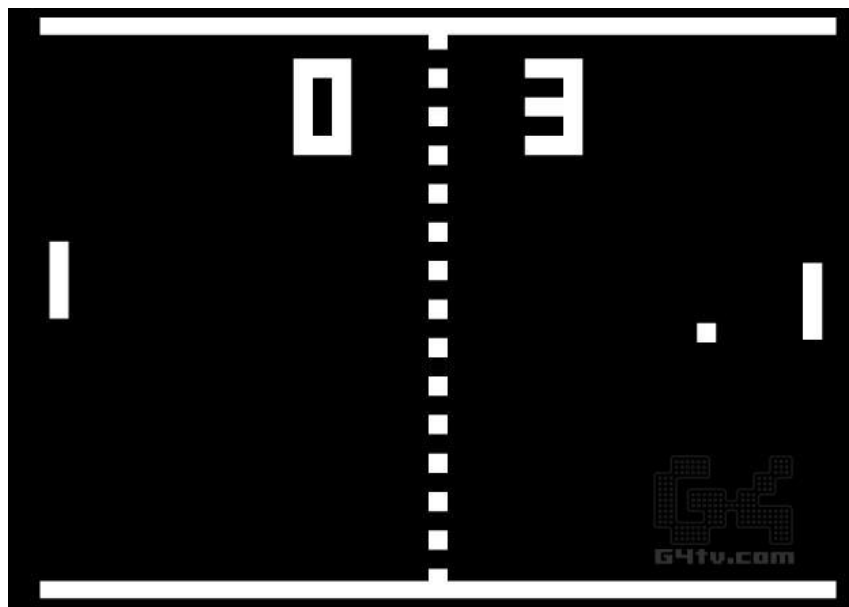
1. Introducción.

Se va a desarrollar un juego que simula el tenis de mesa o ping pong, implementando múltiples funciones utilizando los conceptos de programación orientada a objetos, archivos de texto y estructuras de datos como matrices.

2. Pong.

Pong fue el primer juego desarrollado por Atari Inc, fundada en junio de 1972.

El jugador controla en el juego una paleta moviéndola verticalmente en la parte izquierda de la pantalla, y puede competir tanto contra un oponente controlado por computadora, como con otro jugador humano que controla una segunda paleta en la parte opuesta. Los jugadores pueden usar las paletas para pegarle a la pelota hacia un lado u otro. El objetivo consiste en que uno de los jugadores consiga más puntos que el oponente al finalizar el juego. Estos puntos se obtienen cuando el jugador adversario falla al devolver la pelota.



El juego de Pong que el estudiante va a crear es una herramienta sencilla que permite el entretenimiento de aquellos que la utilicen y que integra:

- Definición de objetos, sus datos y métodos.
- Manejo de unidades de tiempo.
- Generación de eventos aleatorios.
- Gráficos, sonidos y otros
- Trabajo en equipo

3. Descripción Funcional Programa.

La presentación de la interfaz con el usuario es **totalmente libre** y será un elemento importante dentro de la calificación del proyecto.

El proyecto va a ser continuado en el curso (tercer proyecto), haciendo una interfaz con algún hardware específico que permita realizar algunas funciones que se implementarán en este proyecto.

El tablero de juego va a ser considerado una matriz de tamaño $n \times m$, donde n va a ser 25 y m va a ser 40.

Al inicio del juego se va a escoger el modo de juego:

- Humano vs humano
- Humano vs computador

Además se debe escoger el nivel a jugar:

- 1 paleta (single)
- 2 paletas (doubles): en este caso son dos paletas para cada jugador, que son manejadas por el mismo control de movimiento.

La paleta va a iniciar con tamaño 9 donde las primeras 3 posiciones indican golpe hacia arriba, las segundas 3 posiciones indican golpe en línea recta y las 3 posiciones de abajo indican golpe hacia abajo.

El juego consta de 3 niveles. Cada vez que se avanza de nivel, se aumenta la velocidad de la pelota (a definir por los programadores) y se disminuye en 3 el tamaño de las paletas.

El primer paso que deben realizar los programadores es definir un modelo de objetos para el proyecto, tomando en cuenta las consideraciones indicadas anteriormente.

El equipo de trabajo debe utilizar en forma obligatoria la plataforma de desarrollo colaborativo GitHub.

Consideraciones a realizar.

- Establecer una relación tiempo – espacio - acción.
- El modelo de objetos debe documentarse. (investigar)

4. Funciones a investigar.

Debe **investigarse** el uso de algunas funciones referentes a validaciones de datos y despliegue de información. Las funciones que podrían utilizarse, entre otras son:

- Utilización de multimedia: integración de animaciones, sonidos y otros.
- Generación de números aleatorios
- Manejo de archivos de texto

5. Documentación.

La documentación interna en el programa fuente, debe contener antes de definir cada función, al menos una explicación de lo que realiza la función, las entradas, salidas y restricciones consideradas.

La documentación externa debe incluir:

- Tabla de contenidos
- Introducción
- Descripción del problema.
- Diagrama de clases: **INVESTIGAR**.
- Análisis de resultados. (incluyendo corridas de ejemplo)
- Dificultades encontradas: problemas en el desarrollo y que se hizo para corregirlos
- Bitácora de actividades: se deben ir anotando todas las actividades, tipo de actividad, su descripción y duración.
- Estadística de tiempos: un cuadro que muestre un resumen de la Bitácora de Actividades en cuanto las horas **REALES** invertidas. Ejemplo:

FUNCION	Integrante 1	Integrante 2	TOTAL
Análisis de requerimientos	xx horas	xx horas	xx horas
Diseño de la aplicación	xx horas	xx horas	xx horas
Investigación de funciones	xx horas	xx horas	xx horas
Programación	xx horas	xx horas	xx horas
Documentación interna	xx horas	xx horas	xx horas
Pruebas	xx horas	xx horas	xx horas
Elaboración documento	xx horas	xx horas	xx horas
TOTAL	xx horas	xx horas	xx horas

- Conclusión personal (por cada integrante del grupo).

6. Evaluación.

Documentación	15%
Interna	5%
Externa	10%

Resultados (ejecución, eficiencia, manejo correcto de estructuras planteadas, presentación)

Funciones:

Modelo de objetos	10 %
Manejo interfaz (creatividad-presentación)	20 %
Manejo de objetos y matrices	25%
Manejo de juego	33%

7. Aspectos Administrativos.

- El proyecto está estructurado para ser realizado por **2** personas.
- La tarea se debe entregar hasta el día 17/05/2018 hasta las 11:59 pm, en forma electrónica, en un archivo comprimido con los nombres de los estudiantes, que contenga TODO lo necesario para poder ejecutarla. Ese día se asignará una cita de revisión del programa. **No se aceptarán tareas después de la fecha y hora indicadas.** Debe enviarse un archivo readme.txt con la versión de Python a utilizar para la revisión y alguna otra indicación que se considere importante. Se debe enviar al correo tareasintrotaller.ce@gmail.com
- La defensa o revisión del proyecto es indispensable y deben asistir los 2 estudiantes que componen el grupo. En esta revisión se preguntará sobre aspectos relacionados con funcionalidad, así como sobre el código. Los estudiantes deben mostrar TOTAL dominio de estos dos temas, de lo contrario, el proyecto puede ser considerado como una copia.
- En caso de probarse algún tipo de fraude en la elaboración de la tarea se aplicarán todas las medidas indicadas al inicio del curso, incluyendo una carta al expediente del estudiante.
- Se debe incluir en el archivo comprimido la documentación solicitada. Debe entregarse en formato electrónico (archivo .pdf o .odt).
- No se aceptarán tareas cuyo archivo sobrepase 2 mb de espacio en disco.
- Se debe adjuntar la documentación solicitada. Debe entregarse en formato electrónico (archivo .doc o .pdf).
- Cualquier falta a los aspectos aquí enunciados implicará pérdida de puntos.

7. Bibliografía.

Documentación técnica Python

8. Consultas.

Puede dirigir cualquier consulta a jschmidtcr@gmail.com o al grupo del curso en facebook.