

# Introducción a la Programación

## Proyecto de Semestre 2017-1

Grupos B y I

4 de abril de 2017

Road Fighter es un juego de carros desarrollado por Konami para Arcades en 1984. En este juego el objetivo es alcanzar la meta antes que se acabe el tiempo, sin chocar otros carros y sin que se le acabe la gasolina. La idea de este proyecto es desarrollar este juego usando Python con ayuda de alguna de las librerías gráficas existentes (Tkinter o Turtle).



El juego consiste en un carro que se mueve por una carretera y debe llegar a una meta. En el camino puede encontrarse con tres diferentes tipos de carros:

**Minivan:** es un carro que no cambia de carril en ningún momento.

**Runner:** es un carro que constantemente cambia de carril, pero no tiene en cuenta dónde se encuentra el carro del jugador.

**Fighter:** es un carro cuyo objetivo es golpear al carro del jugador y por esto siempre se cambia al carril donde está el carro del jugador.

En la versión *Versus* del juego, dos jugadores juegan al mismo tiempo, en carreteras separadas pero con las mismas condiciones, es decir, el mismo trayecto, aparecen los mismos elementos en la carretera.

El juego termina cuando uno de los dos jugadores ha llegado a la meta, o a ambos se le ha acabado la gasolina.

## 1. Requerimientos

La idea es implementar este juego con base en los siguientes requerimientos:

- El juego se debe desarrollar para dos jugadores, que jueguen al tiempo. Se debe permitir ingresar el nombre de los jugadores, y estos siempre deben aparecer en la pantalla.
- En cualquier momento del juego se debe poder guardar en un archivo el estado para ser cargado en una ejecución posterior del juego.
- La física debe respetarse (e.g. choques, dirección, velocidad, rozamiento).
- Los enemigos pueden tener diferente velocidad y esta debe aumentar dependiendo del nivel de dificultad del juego.
- Al principio del juego debe parametrizarse el nivel de dificultad del juego (mínimo 5 niveles). Todos los niveles deben hacerse de manera que se juegue mínimo un minuto.
- La gasolina de los carros va disminuyendo cada cierta distancia recorrida. Si llega a cero, el carro para y no puede terminar la carrera.
- Los dos carros se manejarán con el teclado.
- En la pantalla se debe mostrar el juego con su estado actual, y los nombres, la velocidad y la cantidad de gasolina de los jugadores.
- Los carros enemigos salen por la parte de arriba de la pantalla y se desplazan hacia la parte inferior. Si un carro enemigo llega a la parte inferior, este desaparece. Si un carro enemigo choca a un jugador, se disminuye su velocidad y cantidad de gasolina.
- Deben haber mínimo tres clases de carros enemigos. Adicional a esto, en la carretera pueden salir manchas de aceite que desestabilizan el carro, es decir, cambian la dirección y velocidad del mismo por un corto tiempo. Estas manchas de aceite deben aparecer en la carretera de manera aleatoria. De igual manera cada cierto tiempo (aleatorio) aparece un galón de gasolina que, si el carro pasa por encima, se suma a la cantidad de gasolina actual.

Este proyecto se debe realizar en el lenguaje de programación Python. No puede utilizar librerías con motores de juego como PyGame o Kivy, ni herramientas de simulación.

## 2. Equipo de trabajo

El proyecto se debe realizar individualmente aunque se pueden reunir para discutir aspectos del proyecto.

### 3. Cronograma

- 19 de abril: Deben entregar un documento donde se hace un análisis y descripción de las reglas del juego (Elementos, relaciones e interacciones), y un prototipo gráfico del juego (puede desarrollarse en cualquier herramienta). Adicionalmente, se debe investigar y presentar un pequeño resumen de la librería gráfica que piensan usar en el juego. Se debe realizar una presentación (máximo 3 minutos, 2 o 3 slides) donde se muestre la información anterior.
- 24 de mayo: Entrega final y sustentación.

### 4. Entregables

Se debe entregar el software (documentado), acompañado de un manual de usuario, en el cual explique cómo instalar y como jugar. El software debe estar en algún manejador de versiones (GitLab, Github, BitBucket, etc.). Adicionalmente se debe entregar un poster sobre el desarrollo del juego, el cual será presentado el día de la sustentación.

### 5. Criterios de evaluación

- La primera entrega del proyecto asume el 10 % de la nota.
- El producto debe cumplir con todos los requerimientos. Cada uno de ellos con la misma ponderación. La totalidad de los requerimientos equivalen al 45 % de la nota.
- El uso correcto de las nociones de programación vistas en clase tiene el 20 % de la nota.
- El manual de usuario, poster y documentación del código suma el 15 % de la nota.
- El 10 % restante es la sustentación en la cual se verificará el desarrollo de todo el proyecto.