



Racing

Mario Martínez Lanuza

Universidad Cenfotec

FIS-01 Física 1

Luis Guillermo Muñoz

Fecha: julio, 2011

Resumen Ejecutivo

Racing, es una aplicación que mezcla el juego con conceptos teóricos de física, específicamente el de Movimiento Rectilíneo Uniformemente Variado, que viene a ser una mezcla entre el Movimiento Rectilíneo Uniforme, y Movimiento Rectilíneo Uniformemente Acelerado.

Es un juego para dos jugadores, donde cada uno dispone de 30 segundos para hacer un recorrido por la pista, en el transcurso de la carrera cada jugador puede acelerar o desacelerar, y el objetivo es hacer el menor tiempo en el recorrido para ganar la partida.

Durante el recorrido se muestra la velocidad que lleva el carro.

El juego fue desarrollado en Processing un aplicación de código abierto.

Palabras clave: velocidad, aceleración, processing, física, mru, mrua.

Tabla de Contenido

Introducción	4
Capítulo I: Justificación teórica del proyecto	5
Velocidad instantánea:.....	5
Velocidad media	6
Aceleración instantánea.....	7
Aceleración promedio	7
Tiempo	7
Capítulo II: Processing	8
Capítulo III: Estructura del programa	8
racing	8
Game	9
Capítulo IV: Figuras y tablas	12
Capítulo V: Lista de referencias	16

Lista de figuras (Solo si las hay)

<i>Figura 1: tiempo previo inicio de jugador 1</i>	<i>12</i>
<i>Figura 2: jugador 1 en juego.....</i>	<i>13</i>
<i>Figura 3: Tiempo previo inicio jugador dos.....</i>	<i>14</i>
<i>Figura 4: fin del juego</i>	<i>15</i>

Introducción

Es presente documento constituye la documentación del proyecto del curso de física 1, desarrollada en Processing, IDE utilizado en el curso con fines de aprendizaje de la física.

El objetivo es reforzar algunos conceptos desarrollados en este curso.

En el primer capítulo I se aborda el tema de la justificación teórica del proyecto, que son, principalmente, velocidad y aceleración.

En el capítulo II se describe el software utilizado para desarrollar el juego, que es Processing.

En el capítulo III se define la estructura de la aplicación; las clases utilizadas, así como el nombre y una breve descripción de los métodos utilizados.

En el capítulo IV se agregan las figuras relativas el juego,

Finalmente, en el capítulo V se agrega la bibliografía utilizada en este proyecto.

Capítulo I: Justificación teórica del proyecto

El presente proyecto se basa en dos conceptos teóricos desarrollados en el curso de Física 1, denominado “Movimiento rectilíneo uniforme¹” y “Movimiento rectilíneo uniformemente acelerado²”. El MRU implica que a lo largo del movimiento la velocidad se mantiene constante, lo que implica que la aceleración es cero, y el MRUA implica que a lo largo del movimiento hay aceleración mayor a cero y esta es constante, lo que implica que la velocidad varía en cada variación del tiempo.

Dado que el objetivo de este proyecto es permitir que cada jugador modifique la aceleración de su carro (acelere o desacelere) en cualquier momento, pero no mantenerlo en todo el recorrido, implica que no se pueden utilizar las ecuaciones de ambos tipos de movimientos en su forma pura, pues en el caso del MRU no se cumple pues se permite acelerar, y en el caso del MRUA, también se incumple, pues el jugador puede mantener la velocidad constante, lo que implica que la aceleración puede ser positiva, negativa o cero.

Por lo tanto, se recurre al concepto teórico de velocidad media y aceleración media, que es una aproximación al concepto teórico de velocidad instantánea y aceleración instantánea, que son:

Velocidad instantánea:

Esta velocidad es la variación en la velocidad ante un cambio en la distancia recorrida entre la fracción de tiempo utilizada para alcanzar dicho desplazamiento, con la característica que la variación en el tiempo tiende a cero, es decir, se utiliza el concepto de límite, de esta forma se calcula así:

¹ Por sus siglas MRU

² Por sus siglas MRUA

$$v = \lim_{\Delta t \rightarrow 0} \frac{\Delta x}{\Delta t} = \lim_{t \rightarrow t_1} \frac{x_1 - x}{t_1 - t}$$

Donde x es la posición en el instante t y x_1 la posición en el instante t_1 .

El límite de una función viene ser una aproximación de la pendiente que se genera al obtener la primera derivada de la función, pero esta requiere que la función sea continua, y en nuestro caso es discreta, pues ya que el cambio en el tiempo generado en el software utilizado entre cada ciclo es aproximadamente de 44 milésimas de segundos.

Por esta razón se utiliza la ecuación de velocidad media

Velocidad media

La velocidad media es el cociente del incremento en el desplazamiento y el incremento en el tiempo, que en este caso, como se indicó en el apartado anterior es de 44 milésimas de segundo, está dada por la ecuación:

$$v_m = \frac{\Delta x}{\Delta t} = \frac{x_1 - x_0}{t_1 - t_0}$$

Donde x_0 es la posición en el instante inicial t_0 , y x_1 la posición en el instante final t_1 .

De esta forma se puede acelerar en un instante, incrementando el rango de desplazamiento por fracción de tiempo, modificando de esta forma la velocidad, y al dejar de acelerar o desacelerar, el desplazamiento por fracción de tiempo es constante, permitiendo de esta forma observar que la velocidad se mantiene constante.

Adicionalmente se utiliza esta ecuación para generar la velocidad media de todo el recorrido para cada jugador.

Una situación similar sucede con la aceleración, por lo que se utiliza la aceleración instantánea.

Aceleración instantánea

De forma similar de cómo se calcula la velocidad instantánea, en este caso también se utiliza el concepto de derivada o su aproximación de límite, siendo la ecuación la siguiente:

$$a = \lim_{\Delta t \rightarrow 0} \frac{\Delta v}{\Delta t} = \lim_{t \rightarrow t_1} \frac{v_1 - v}{t_1 - t}$$

Donde v es la velocidad en el instante inicial t y v_1 la velocidad final en el instante t_1 .

Y, nuevamente, dado que el ciclo del software genera una función discreta se utiliza su aproximación que corresponde a la aceleración promedio.

Aceleración promedio

La aceleración media es una aproximación a la aceleración instantánea, ya que como se indicó anteriormente los ciclos son en promedio cada 44 milisegundos; su ecuación es:

$$a_m = \frac{\Delta v}{\Delta t} = \frac{v_1 - v_0}{t_1 - t_0}$$

Donde v_1 es la velocidad en el instante inicial t_0 y v_0 la velocidad final en el instante t_0 .

Tiempo

Se utiliza una función que acumula el tiempo utilizando fracciones en milisegundos para cada jugador, esta se llama 'millis()' del software "Processing".

El juego guarda el tiempo que cada jugador consumió para hacer el recorrido a la pista, y es este el que se utiliza para identificar al ganador.

En el siguiente apartado se describen las características del software utilizado en el proyecto.

Capítulo II: Processing

Processing es un lenguaje de programación y entorno de desarrollo de código abierto basado en Java³, que permite escribir, editar, compilar y ejecutar código java⁴. Fue iniciado por Ben Fry y Casey Reas, ambos miembros de Aesthetics and Computation Group del MIT Media Lab dirigido por John Maeda⁵.

Según lo declara Processing Funtation⁶, su misión es promover la alfabetización en software dentro de las artes visuales y la alfabetización visual dentro de los campos relacionados con la tecnología, haciendo que estos sean accesibles a diversas comunidades. Para que personas de todos los intereses y orígenes adquieran la capacidad de programar y hacer sus trabajos creativos.

La documentación, así como otros link como bibliotecas, foros, etc, se pueden ubicar en la página <https://processing.org/>.

Capítulo III: Estructura del programa

El programa se realiza en el IDE de Processing, y se utilizaron dos capas, que son Racing y Game.

racing

La clase racing contiene las variables globales, su inicialización (setup), el bucle draw y el método keyPressed.

El método draw carga para cada frame lo siguiente:

- a) El fondo o background

³ Wikipedia, Processing

⁴ Air Room, PROCESSING: 1.5 ¿Qué es Processing?

⁵ Processing 2⁰ Bachillerato.

⁶ Processing Foundation

- b) La imagen del carro, la coordenada en el eje x, la coordenada en el eje y, el largo y ancho del carro, el mensaje del control del tiempo, la velocidad media de cada carro, así como la aceleración media de cada carro, todas que provienen de métodos de la clase Game.

El método keyPressed, en el cual se habilita el uso de las siguientes teclas:

- a) up, down, right, left, que permiten al jugador utilizar como direccionamiento.
- b) Las letras 'a', 'A' que permite al jugador acelerar el carro.
- c) Las letras 'd', 'D' que permite al jugador desacelerar el carro.
- d) s imágenes que se deben mostrar, así como la posición en el eje 'x' y 'y'

Por otro lado, la clase Game, contiene los siguientes métodos

Game

Esta clase contiene toda la lógica del juego, con los siguientes métodos:

- 1) setCoordX, es el setter de la coordenada x,
- 2) setCoordY, es el setter de la coordenada y,
- 3) cargarImágenes, carga las imágenes que se utilizan en el juego,
- 4) inicializarPrimerasImágenes, inicializa las imágenes que se requieren al iniciar el juego,
- 5) controlTiempo, este retorna una variable de tipo "string" que contiene la información del estado del juego y el cronómetro.
- 6) Play, permite según el momento (tiempo) de juego invocar los métodos:
 - (1) inicializarInicio
 - (2) inicializarSalidaJugador1
 - (3) inicializarJugador2
 - (4) inicializarSalidaJugador2

- 7) inicializarInicio, permite inicializar las imagenes del carro y del fondo al inicio del juego,
- 8) inicializarSalidaJugador1, permite mantener el carro del jugador 1, en la posición de salida,
- 9) inicializarJugador2, permite inicializar las variables para la salida del jugador 2,
- 10) inicializarSalidaJugador2, permite mantener el carro del jugador 2, en la posición de salida,
- 11) aceleracionCar, permite acelerar, según el estado en que se encuentre el carro,
- 12) desaceleracionCar, permite desacelerar, según el valor de aceleración previo que presente el carro,
- 13) giroDerecha, permite girar el carro a la derecha, estableciendo los valores de las coordenadas 'x' y 'y' según el sentido del desplazamiento,
- 14) giroIzquierda, Método que permite girar el carro a la izquierda, estableciendo los valores de las coordenadas 'x' y 'y' según el sentido del desplazamiento,
- 15) giroArriba, permite girar el carro hacia arriba, estableciendo los valores de las coordenadas 'x' y 'y' según el sentido del desplazamiento,
- 16) giroAbajo, Método que permite girar el carro hacia abajo, estableciendo los valores de las coordenadas 'x' y 'y' según el sentido del desplazamiento,
- 17) desplazamiento, permite establecer el valor de la coordenada del desplazamiento, según el sentido del desplazamiento,
- 18) velocidadMedia, Método que permite establecer la velocidad media por ciclo. Este método utiliza una constante igual a 22.315 que proviene de 44.8129 ciclos por cada segundo.valor aproximado, ya que los ciclos de la función draw no son constantes, provocando que pese a no modificar la aceleración el valor de la velocidad sea diferente en cada ciclo. Con esto se logra que la velocidad sea constante cuando no

se acelere o desacelere, y es una aproximación al valor generado como velocidad promedio de todo el recorrido.

19) `aceleracionMedia`, Método que permite establecer la aceleración media por ciclo, se utiliza la misma constante del punto anterior, por el mismo razonamiento.

20) `recorridoTotal`, permite establecer recorrido total de la carrera,

21) `stopRace`, permite detener la carrera, cuando el carro hace el recorrido total de la pista.

Capítulo IV: Figuras y tablas

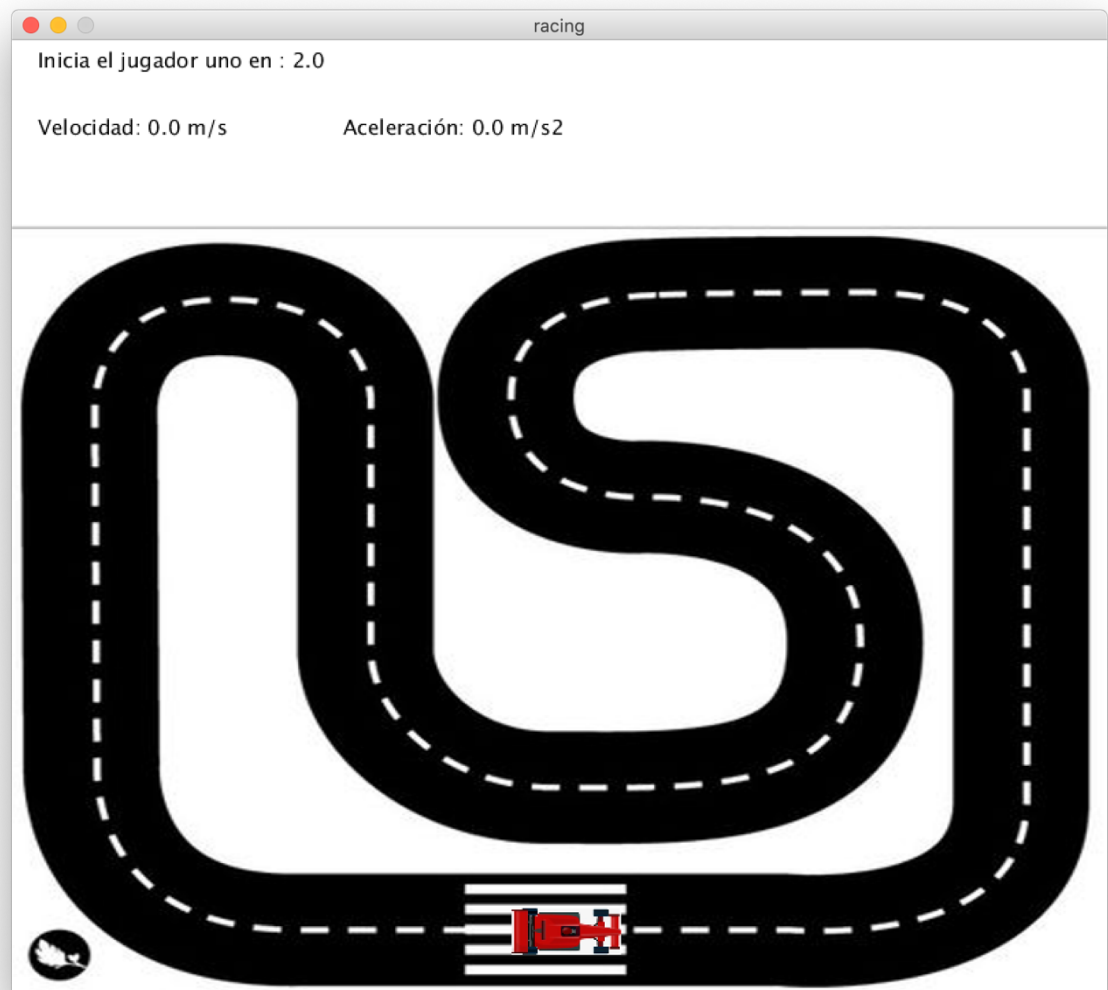


Figura 1: tiempo previo inicio de jugador 1

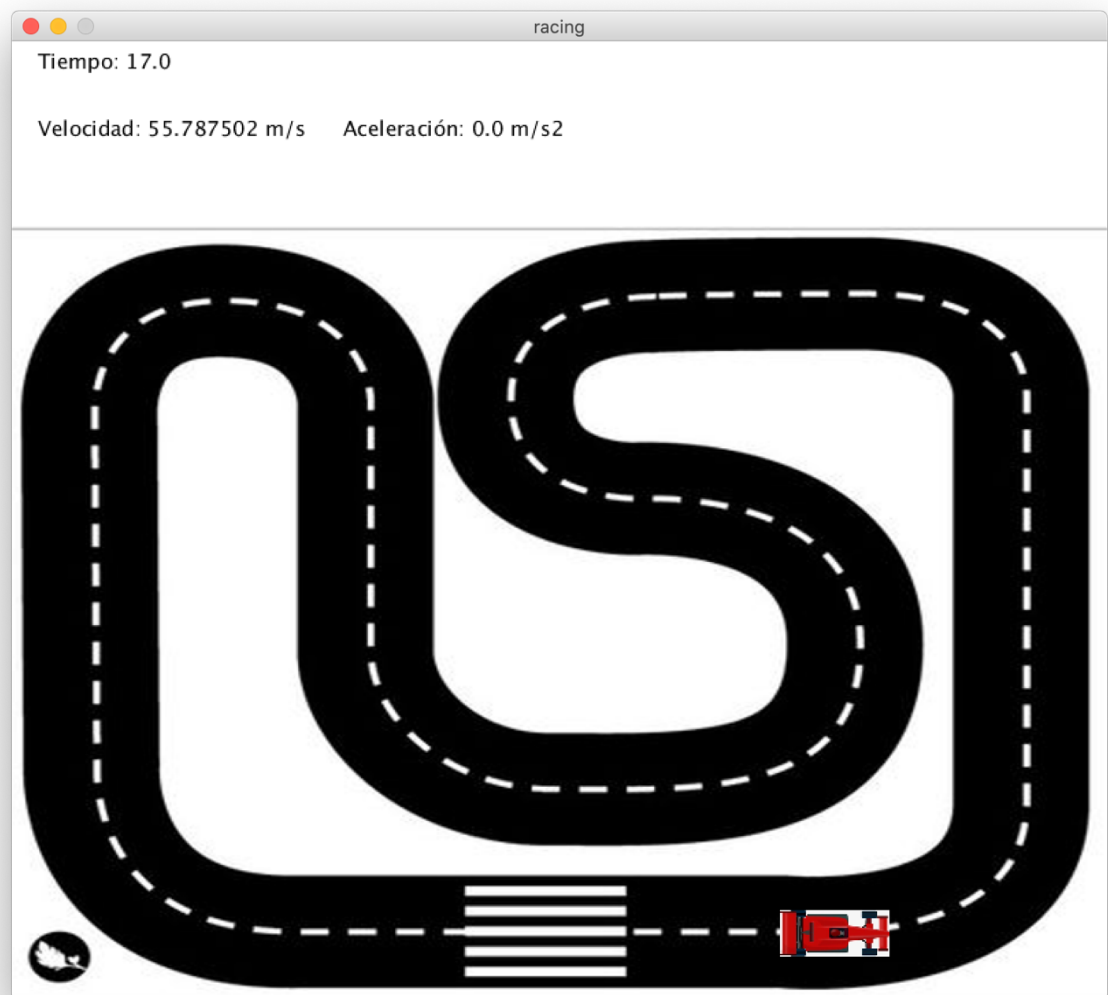


Figura 2: jugador 1 en juego

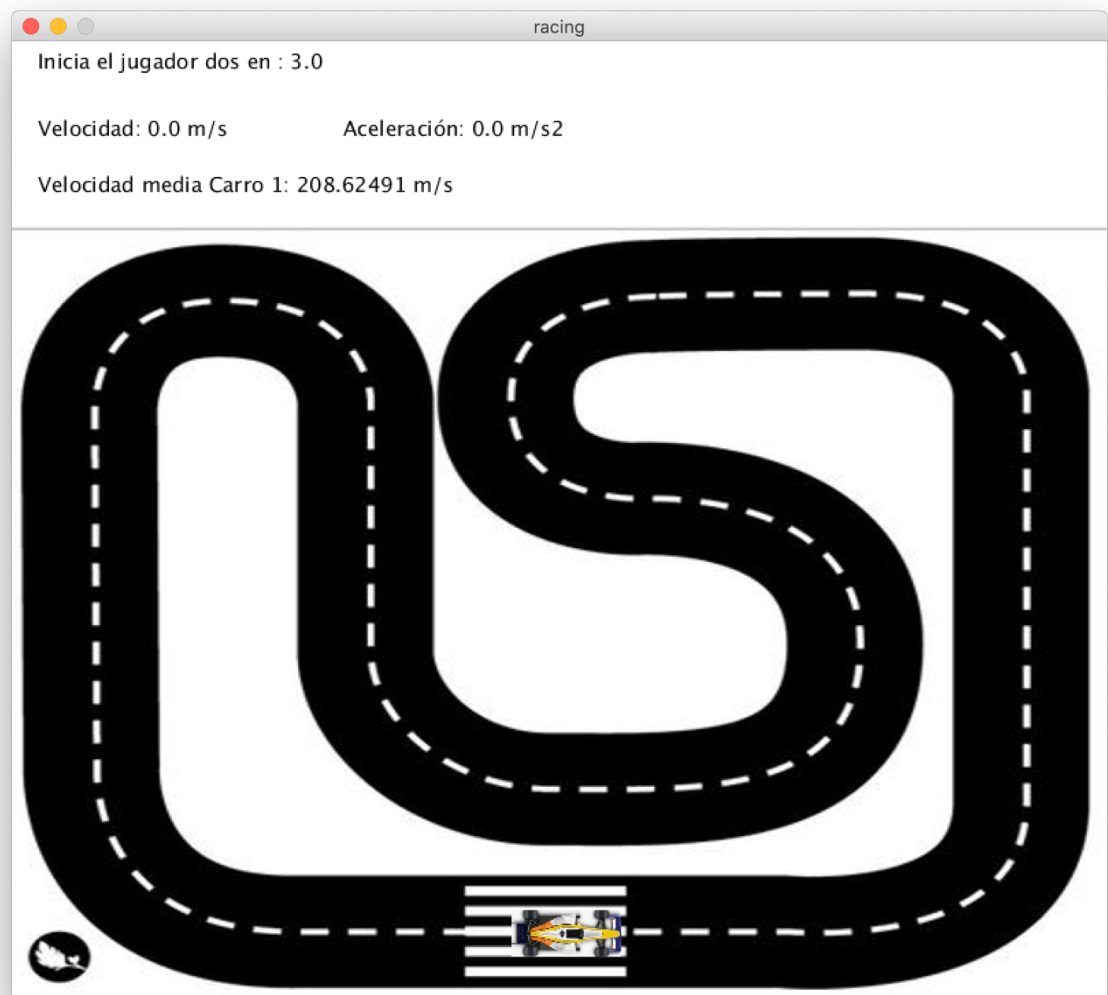


Figura 3: Tiempo previo inicio jugador dos

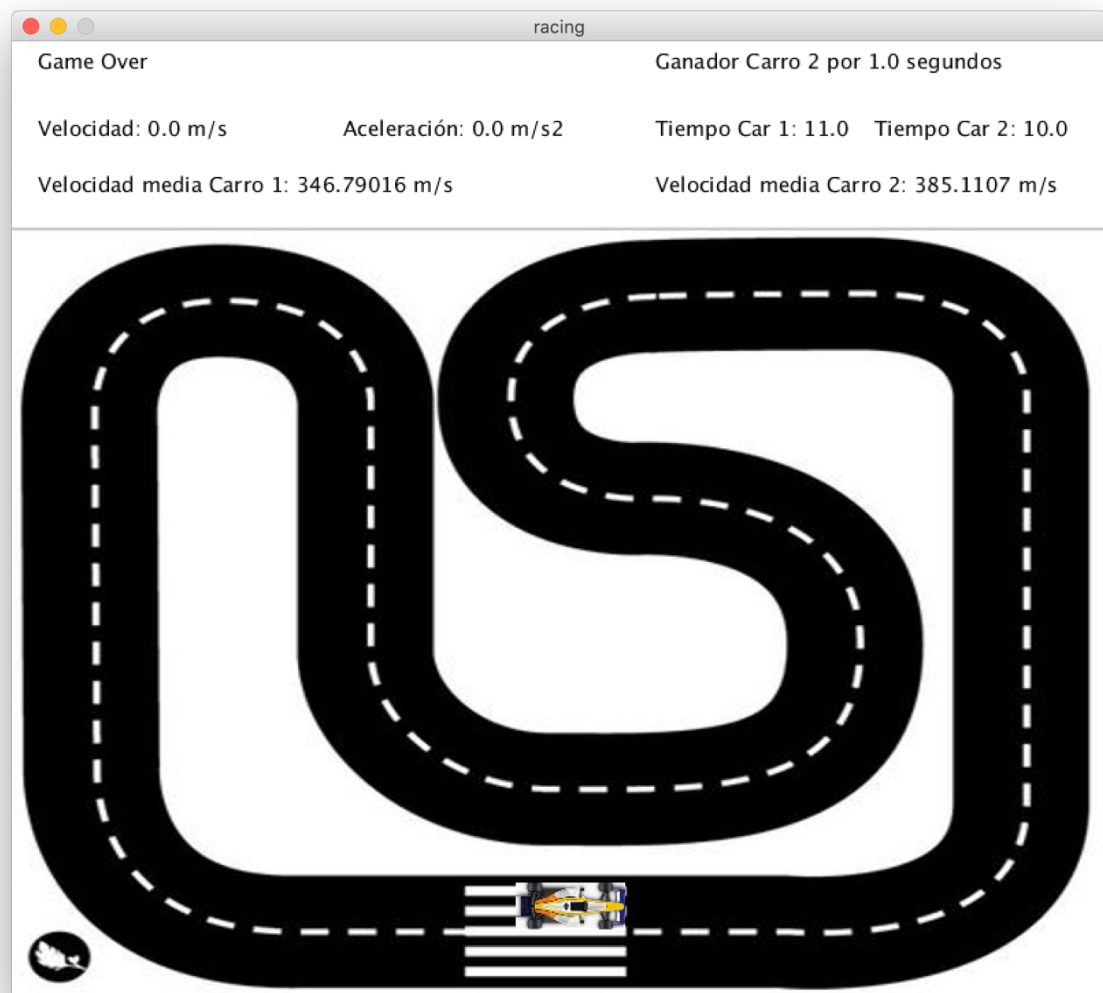


Figura 4: fin del juego

Capítulo V: Lista de referencias

Wikipedia, Processing, recuperado el 24/07/2021 de: <https://es.wikipedia.org/wiki/Processing>

Air Room, PROCESSING: 1.5 ¿Qué es Processing?, YouTube, recuperado el 24/07/2021 de:

<https://www.youtube.com/watch?v=7dYiN1tMjxk>

Processing 2⁰ Bachillerato, recuperado el 24/07/2021 de:

<https://sites.google.com/site/processing2obachillerato/>

Processing.org; recuperado el 24/07/2021 de: <https://processing.org/>

Processing Foundation, recuperado el 24/07/2021 de: <https://processingfoundation.org/>