

Nivel Secundario

Programación en Python - Desafío de la Ronda 1

Bienvenidos a la Olimpiada Federal de Programación y Robótica - OFIRCA 2025. Estamos muy contentos de que participen del evento. Vamos a compartir momentos de aprendizaje y mucha diversión. ¿Listos para activar su creatividad? ¡¡Comencemos!!

Introducción

En un mundo cada vez más conectado, la inteligencia artificial comienza a jugar un papel fundamental en nuestras vidas. Las ciudades comienzan a ser cada vez más inteligentes: vehículos autónomos, semáforos que se adaptan al tráfico en tiempo real, asistentes virtuales que ayudan en hospitales y sistemas que optimizan el consumo de energía. Todo esto es posible gracias al trabajo de personas que entienden cómo funciona la tecnología y saben cómo aplicarla para resolver problemas del mundo real. En este contexto, la programación se convierte en una herramienta clave para construir el futuro. No se trata solo de escribir código, sino de pensar de manera lógica, creativa y estratégica. Participar en esta competencia requiere ganas de aprender, de resolver problemas y de divertirse en el camino. La OFIRCA del presente año tiene como objetivo fomentar el desarrollo de habilidades de programación entre los jóvenes promoviendo el uso de la inteligencia artificial y reflexionando sobre las ciudades inteligentes.

Objetivo general de la actividad

Esta misión consiste en la mejora del desarrollo de un juego interactivo basado en una historia en la que los personajes UAIBOT, UAIBOTA y sus hijos UAIBOTINA y UAIBOTINO recorren una ciudad para realizar entregas.

Los proyectos ganadores de esta olimpiada serán compartidos con estudiantes de escuelas de nivel primario. Buscamos que cada juego pueda ser recorrido e intervenido por quienes quieran contribuir a enriquecer la propuesta, y crear nuevas ideas a partir de lo que programaron.

Características del juego

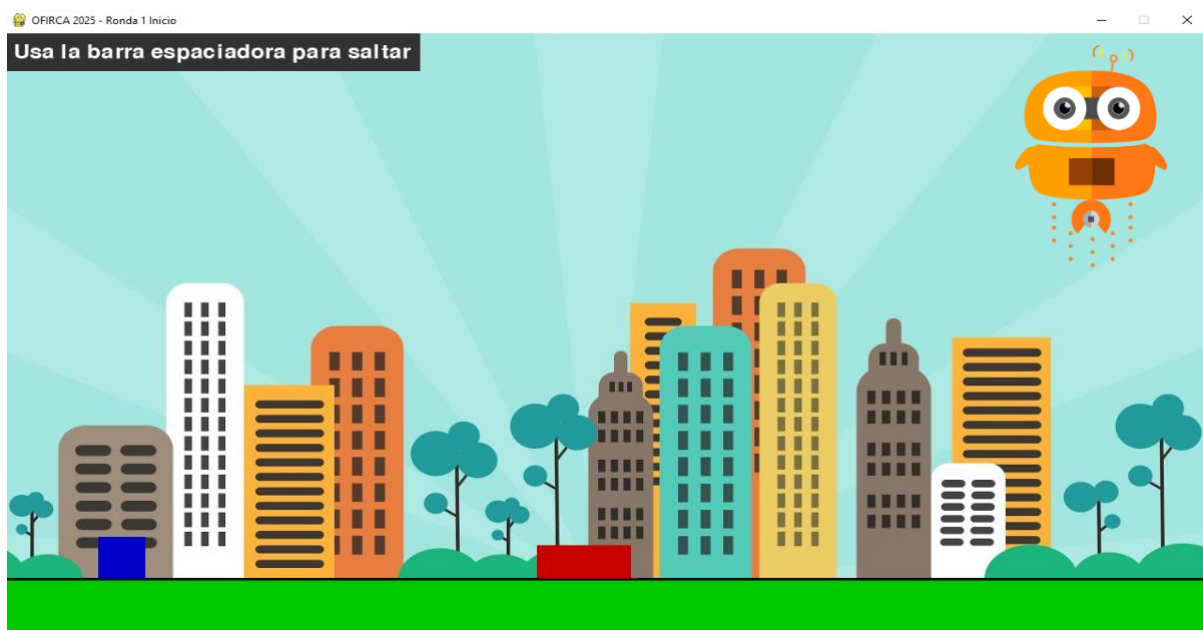
- Se muestra un escenario urbano en donde navegarán los robots.
- El juego está realizado en dos dimensiones con vista de costado.
- Al comenzar el juego se muestran las reglas en pantalla.

Requerimientos sobre los cuales se basó el juego que les facilitamos

Con objeto de facilitarles la lectura y la comprensión del código del juego, les detallamos los requerimientos sobre los cuales se basó su arquitectura:

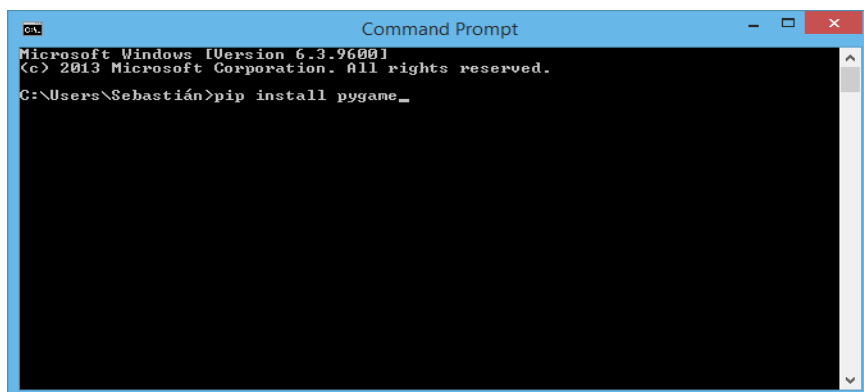
1. En la etapa inicial se realizó:
 - 1.1. El desarrollo del código que permite:
 - 1.1.1. Mostrar una ciudad como fondo de pantalla.
 - 1.1.2. Presentar las reglas del juego en su interfaz gráfica.
2. En la etapa avanzada se realizó:
 - 2.1. El desarrollo del código que permite:
 - 2.1.1. Mostrar el avatar de nuestro personaje representado como un cuadrado.
 - 2.1.2. Mostrar un auto que se desplaza sobre un camino en dirección a nuestro personaje representado como un rectángulo rojo.

A continuación, se ilustra lo realizado:



Para realizar los requerimientos utilizamos funciones de la librería Pygame. La misma es fundamental para ejecutar el código provisto y debe instalarse una vez que tengan Python 3.12.3 en sus computadoras de la siguiente manera:

Verificar que poseen una conexión a Internet activa. Abrir en Windows la aplicación Símbolo del Sistema y ejecutar pip install Pygame.



```
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\Sebastián>pip install pygame_
```



Una vez instalada la librería Pygame, quedará en su sistema y la misma les servirá para todas las rondas.

Consignas específicas de trabajo

A partir del juego que le facilitamos, incorporen los siguientes ajustes al juego:

1. Reemplazar al cuadrado azul con la imagen provista de UAIBOT o similar y al rectángulo rojo por la imagen provista del auto o similar. No es necesario que se respeten las proporciones del cuadrado y rectángulo.
2. Animar el fondo para generar una sensación de velocidad. Se puede utilizar el mismo o se puede cambiar.
3. Agregar que UAIBOT salte para esquivar el auto utilizando la barra espaciadora.
4. Si el auto sale de los límites del escenario, deberá aparecer en la posición del inicio.
5. Agregar una barra de energía que dure 60 segundos y se vaya decrementando mientras muestra el porcentaje restante dentro de la misma.
6. Agregar un contador de kilómetros restantes que UAIBOT recorre para llevar un paquete El contador de kilómetros se decrementará 0.03km. cada segundo. La cantidad de kilómetros que UAIBOT debe recorrer es 1km.
7. Si UAIBOT choca con un auto tiene que aparecer un texto de “JUEGO TERMINADO” pausando el juego. En cambio, si recorre un kilómetro, aparece un texto de “¡El paquete fue entregado con éxito!”
8. Se fomenta el uso de Inteligencia Artificial para realizar los desafíos. En caso de usarla, agregar un archivo de texto .txt a la solución con los prompts utilizados.

9. Cualquier funcionalidad extra sumará muchos puntos en la evaluación del jurado siempre y cuando estén cumplidas las consignas anteriores. El jurado presta mucha atención a la creatividad. Indicar en un archivo de texto qué se agregó.
10. Cuando finalicen las consignas del desafío guarden su proyecto de manera local y luego, suban el proyecto en la "Plataforma de OFIRCA" en formato ZIP, comprimiendo todos los archivos necesarios para su corrección. ¡Éxitos!

Conceptos y herramientas que pueden utilizar

- IDE para Python 3.12.3 IDLE (entorno de desarrollo local).
- Uso de librerías.
- Ingreso de datos por teclado
- Uso de estructuras secuenciales, condicionales (if..else) y de repetición (while, for).
- Estructuras de datos
- Definición y uso de funciones
- Uso de imágenes y teclado para animar objetos

Recursos multimedia

Para la realización de esta actividad se podrá utilizar material multimedia generado por el equipo participante, o aquel que cuente con derechos de autor con licencia Creative Commons que permita su reutilización de forma gratuita.

Criterios de evaluación

- **Respeto por las consignas** sobre las cuales se basó el juego que les facilitamos.
- **Legibilidad** en los **códigos** y **claridad** en los **comentarios** representativos de la funcionalidad o tarea que están describiendo.
- **Claridad en la definición de los nombres** y significados de las **variables** y **funciones**.
- **Coherencia estética y facilidad de uso del juego.**
- **Creatividad y originalidad en las propuestas de soluciones**
- **Uso responsable de la Inteligencia Artificial.**
- **Documentación del uso de la Ingeniería de Prompts.**