



## **Simulación**

**Tema:** Inteligencia Artificial 1.



## **Examen Final**

### Objetivo:

- Consolidar los conocimientos adquiridos en clase sobre la Inteligencia Artificial (IA) aplicada a juegos y búsquedas.

### Enunciado:

1. Validar y probar la librería de GYM con python utilizando algunos de los siguientes ejemplos  
[https://gym.openai.com/envs/#toy\\_text](https://gym.openai.com/envs/#toy_text)

<https://towardsdatascience.com/reinforcement-learning-with-openai-d445c2c687d2>

2. Describir paso a paso y comentado el código, realizar varias casos de pruebas y ejecutar.
3. Dentro del juego el usuario deberá escoger/ingresar su ciudad natal incluido latitud y longitud y en base a ello recomendar usuarios cercanos utilizando el algoritmos A\* y Yenn, se debe tener una base de datos de al menos 50 usuarios dentro de una misma ciudad (Tomar datos de pruebas anteriores o generar una nueva base de datos), tener en presente que el árbol debe tener al menos 7 niveles o superior y con 3 conexiones de nodos cada uno.
4. En base a la información proporcionada se deberá generar un sistema que permita mostrar usuarios cercanos y recomendar usuarios con los que se debe jugar o conocer.
5. Realizar el sistema con una interfaz gráfica para acceder y probar el sistema.

Código y documentos de entrega: Se deberá entregar un informe con el procesos dentro del mismo tener capturas del uso del juego y generar un documento en PDF de validación y pruebas. Finalmente subir todo al repositorio incluido los códigos fuentes y los resultados de grafos y el juego.

### **Criterios de Evaluación:**

- Neo4J y Búsquedas : 30%
- Juego IA: 30%
- GUI: 20%
- Informe PDF: 20%
- Usabilidad: 10%

Fecha de entrega: 02/08/2021 – 23:55.

**Nota:** Cualquier pregunta o duda con respecto al examen escribirme por correo electrónico o whatsapp.



## Simulación

Tema: Inteligencia Artificial 1.

## Examen Final

Nombre: Carlos Alvarez

1. Validar y probar la librería de GYM con python utilizando algunos de los siguientes ejemplos

```
(West)
+-----+
|R: | : :G|
| : | : : |
| : : : : |
|█| : | : |
|Y| : |B: |
+-----+

(South)
+-----+
|R: | : :G|
| : | : : |
| : : : : |
|█| : | : |
|Y| : |B: |
+-----+

(South)
Score 7
Score Fínal: 8.35
```

2. Describir paso a paso y comentado el código, realizar varias casos de pruebas y ejecutar

```
#EJEMPLO DE TAXI-V3
#En este entorno hay 4 localizaciones nombradas por diferentes letras (R, G, Y, B), el objetivo es que el taxi (celda
#amarilla en la figura) tiene que coger al pasajero marcado de color Azul (R en este caso) y
#llevarlo a su destino (B, marcado de color violeta en este caso). Una vez el taxi suelta al pasajero
#en su destino, la tarea se ha completado con éxito y el episodio termina

#Cada vez que lo consigamos obtendremos una recompensa de 20 puntos por cada pasajero que logremos llevar a su destino,
#y -1 por cada paso que damos durante el trayecto. También existe una penalización de -10 puntos si se recoge o se deja
#a un pasajero de forma ilegal.
def juego():
    #Empiezo el juego, donde el env busca el juego "Taxi-v3" en la librería gym
    auxScore = 0
    env = gym.make("Taxi-v3")
    env.render()
    action_size = env.action_space.n
    state_size = env.observation_space.n
    qtable = np.zeros((state_size, action_size))
    #Definición de variables para el juego
    total_episodes = 50000      # Total Episodios
    total_test_episodes = 100   # Total Test episodios
    max_steps = 99              # Maximo pasos por episodio

    learning_rate = 0.7         # Tasa de Aprendizaje
    gamma = 0.618               # Tasa de Descuento

    # Parametros de Exploracion
    epsilon = 1.0               # Tasa de Exploracion
    max_epsilon = 1.0           # Probabilidad de Exploracion al inicio
    min_epsilon = 0.01          # Probabilidad minima de Exploracion
    decay_rate = 0.01           # Tasa de decaimiento exponencial para el problema de exploracion
```



**Examen Final**

3.

- Creación de los Datos y Relaciones

```
CREATE (a:Pais {nombre: 'Ecuador'}),
(a1:USUARIO {nombre: 'ANTONIO', latitud: '-21.833', longitud: '-798.833'}),
(a2:USUARIO {nombre: 'MANUEL', latitud: '-2.186', longitud: '-785.097'}),
(a3:USUARIO {nombre: 'JOSE', latitud: '-28.974', longitud: '-790.045'}),
(a4:USUARIO {nombre: 'FRANCISCO', latitud: '-2.542', longitud: '-791.719'}),
(a5:USUARIO {nombre: 'DAVID', latitud: '-32.667', longitud: '-799.667'}),
(a6:USUARIO {nombre: 'JUAN', latitud: '-21.733', longitud: '-798.311'}),
(a7:USUARIO {nombre: 'JAVIER', latitud: '-9.500', longitud: '-807.162'}),
(a8:USUARIO {nombre: 'JOSE ANTONIO', latitud: '-10.544', longitud: '-804.544'}),
(a9:USUARIO {nombre: 'DANIEL', latitud: '-39.906', longitud: '-792.050'}),
(a10:USUARIO {nombre: 'JOSE LUIS', latitud: '-12.417', longitud: '-786.197'}),
(a11:USUARIO {nombre: 'FRANCISCO JAVIER', latitud: '9.500', longitud: '-796.667'}),
(a12:USUARIO {nombre: 'CARLOS', latitud: '-10.333', longitud: '-794.500'}),
(a13:USUARIO {nombre: 'JESUS', latitud: '-16.743', longitud: '-786.483'}),
(a14:USUARIO {nombre: 'ALEJANDRO', latitud: '-21.347', longitud: '-795.872'}),
(a15:USUARIO {nombre: 'MIGUEL', latitud: '3.628', longitud: '-781.300'}),
(a16:USUARIO {nombre: 'JOSE MANUEL', latitud: '-18.167', longitud: '-795.167'}),
(a17:USUARIO {nombre: 'RAFAEL', latitud: '-3.344', longitud: '-784.475'}),
(a18:USUARIO {nombre: 'MIGUEL ANGEL', latitud: '-13.306', longitud: '-785.428'}),
(a19:USUARIO {nombre: 'PABLO', latitud: '-18.667', longitud: '-799.833'}),
(a20:USUARIO {nombre: 'PEDRO', latitud: '-9.333', longitud: '-786.167'}),
(a21:USUARIO {nombre: 'ANGEL', latitud: '8.117', longitud: '-777.186'}),
(a22:USUARIO {nombre: 'SERGIO', latitud: '-6.833', longitud: '-801.000'}),
(a23:USUARIO {nombre: 'JOSE MARIA', latitud: '-33.269', longitud: '-798.049'}),
(a24:USUARIO {nombre: 'FERNANDO', latitud: '-34.522', longitud: '-799.617'}),
(a25:USUARIO {nombre: 'JORGE', latitud: '844', longitud: '-768.911'}),
(a26:USUARIO {nombre: 'LUIS', latitud: '-34.803', longitud: '-802.317'}),
(a27:USUARIO {nombre: 'ALBERTO', latitud: '-10.500', longitud: '-806.667'}),
(a28:USUARIO {nombre: 'ALVARO', latitud: '-19.611', longitud: '-797.256'}),
(a29:USUARIO {nombre: 'JUAN CARLOS', latitud: '-4.625', longitud: '-769.842'}),
(a30:USUARIO {nombre: 'ADRIAN', latitud: '-13.486', longitud: '-805.786'}),
(a31:USUARIO {nombre: 'DIEGO', latitud: '-22.267', longitud: '-808.583'}),
(a32:USUARIO {nombre: 'JUAN JOSE', latitud: '2.333', longitud: '-782.667'}),
(a33:USUARIO {nombre: 'RAUL', latitud: '439', longitud: '-781.561'}),
(a34:USUARIO {nombre: 'IVAN', latitud: '-8.932', longitud: '-794.907'}),
(a35:USUARIO {nombre: 'JUAN ANTONIO', latitud: '-14.500', longitud: '-794.700'}),
(a36:USUARIO {nombre: 'RUBEN', latitud: '-10.439', longitud: '-796.383'}),
(a37:USUARIO {nombre: 'ENRIQUE', latitud: '-24.278', longitud: '-793.389'}),
```



**Examen Final**

(a38:USUARIO {nombre: 'OSCAR', latitud: '-22.083', longitud: '-809.681'}),  
(a39:USUARIO {nombre: 'RAMON', latitud: '-27.333', longitud: '-788.333'}),  
(a40:USUARIO {nombre: 'ANDRES', latitud: '-15.500', longitud: '-797.333'}),  
(a41:USUARIO {nombre: 'VICENTE', latitud: '69', longitud: '-793.958'}),  
(a42:USUARIO {nombre: 'JUAN MANUEL', latitud: '3.272', longitud: '-794.689'}),  
(a43:USUARIO {nombre: 'SANTIAGO', latitud: '-13.600', longitud: '-799.000'}),  
(a44:USUARIO {nombre: 'JOAQUIN', latitud: '-21.667', longitud: '-794.653'}),  
(a45:USUARIO {nombre: 'VICTOR', latitud: '-26.736', longitud: '-796.183'}),  
(a46:USUARIO {nombre: 'CARMEN', latitud: '-14.830', longitud: '-779.870'}),  
(a47:USUARIO {nombre: 'ANA MARIA', latitud: '-16.056', longitud: '-790.031'}),  
(a48:USUARIO {nombre: 'JOSEFA', latitud: '-9.417', longitud: '-792.347'}),  
(a49:USUARIO {nombre: 'MARIA PILAR', latitud: '-9.890', longitud: '-778.159'}),  
(a50:USUARIO {nombre: 'ISABEL', latitud: '1.2864', longitud: '-788.353'}),  
(a51:USUARIO {nombre: 'LAURA', latitud: '-39.833', longitud: '-793.500'}),  
(a52:USUARIO {nombre: 'MARIA DOLORES', latitud: '719', longitud: '-800.508'}),  
(a53:USUARIO {nombre: 'MARIA TERESA', latitud: '3.325', longitud: '-782.136'}),  
(a54:USUARIO {nombre: 'ANA', latitud: '-6.000', longitud: '-804.250'}),  
(a55:USUARIO {nombre: 'CRISTINA', latitud: '1.272', longitud: '-792.531'}),  
(a56:USUARIO {nombre: 'MARTA', latitud: '-18.167', longitud: '-802.333'}),  
(a57:USUARIO {nombre: 'MARIA ANGELES', latitud: '-23.000', longitud: '-781.167'}),  
(a58:USUARIO {nombre: 'LUCIA', latitud: '-21.200', longitud: '-796.900'}),  
(a59:USUARIO {nombre: 'FRANCISCA', latitud: '-9.525', longitud: '-793.531'}),  
(a60:USUARIO {nombre: 'MARIA ISABEL', latitud: '-36.806', longitud: '-796.806'}),  
(a61:USUARIO {nombre: 'MARIA JOSE', latitud: '6.100', longitud: '-778.400'}),  
(a62:USUARIO {nombre: 'ANTONIA', latitud: '8.667', longitud: '-798.333'}),  
(a63:USUARIO {nombre: 'DOLORES', latitud: '-21.900', longitud: '-788.500'}),  
(b1:Ciudad {nombre: 'Guayaquil'}),  
(b2:Ciudad {nombre: 'Quito'}),  
(b3:Ciudad {nombre: 'Cuenca'}),  
(b1)-[:Pertenece]->(a),  
(b2)-[:Pertenece]->(a),  
(b3)-[:Pertenece]->(a),  
(a1)-[:Vive]->(b3),  
(a2)-[:Vive]->(b3),  
(a3)-[:Vive]->(b3),  
(a4)-[:Vive]->(b3),  
(a5)-[:Vive]->(b3),  
(a6)-[:Vive]->(b3),  
(a7)-[:Vive]->(b3),  
(a8)-[:Vive]->(b3),  
(a9)-[:Vive]->(b3),



## Simulación

Tema: Inteligencia Artificial 1.

### Examen Final

(a10)-[:Vive]->(b3),  
(a11)-[:Vive]->(b3),  
(a12)-[:Vive]->(b3),  
(a13)-[:Vive]->(b3),  
(a14)-[:Vive]->(b3),  
(a15)-[:Vive]->(b3),  
(a16)-[:Vive]->(b3),  
(a17)-[:Vive]->(b3),  
(a18)-[:Vive]->(b3),  
(a19)-[:Vive]->(b3),  
(a20)-[:Vive]->(b3),  
(a21)-[:Vive]->(b3),  
(a22)-[:Vive]->(b3),  
(a23)-[:Vive]->(b3),  
(a24)-[:Vive]->(b3),  
(a25)-[:Vive]->(b3),  
(a26)-[:Vive]->(b3),  
(a27)-[:Vive]->(b3),  
(a28)-[:Vive]->(b3),  
(a29)-[:Vive]->(b3),  
(a30)-[:Vive]->(b3),  
(a31)-[:Vive]->(b3),  
(a32)-[:Vive]->(b3),  
(a33)-[:Vive]->(b3),  
(a34)-[:Vive]->(b3),  
(a35)-[:Vive]->(b3),  
(a36)-[:Vive]->(b3),  
(a37)-[:Vive]->(b3),  
(a38)-[:Vive]->(b3),  
(a39)-[:Vive]->(b3),  
(a40)-[:Vive]->(b3),  
(a41)-[:Vive]->(b3),  
(a42)-[:Vive]->(b3),  
(a43)-[:Vive]->(b3),  
(a44)-[:Vive]->(b3),  
(a45)-[:Vive]->(b3),  
(a46)-[:Vive]->(b3),  
(a47)-[:Vive]->(b3),  
(a48)-[:Vive]->(b3),  
(a49)-[:Vive]->(b3),  
(a50)-[:Vive]->(b3),



## Simulación

Tema: Inteligencia Artificial 1.

### Examen Final

(a51)-[:Vive]->(b2),  
(a52)-[:Vive]->(b2),  
(a53)-[:Vive]->(b2),  
(a54)-[:Vive]->(b1),  
(a55)-[:Vive]->(b1),  
(a56)-[:Vive]->(b1),  
(a57)-[:Vive]->(b2),  
(a58)-[:Vive]->(b2),  
(a59)-[:Vive]->(b2),  
(a60)-[:Vive]->(b2),  
(a61)-[:Vive]->(b1),  
(a62)-[:Vive]->(b1),  
(a63)-[:Vive]->(b1)

- Ingreso de los datos del usuario

tk

Ciudad:

Usuario:

Latitud:

longitud:

Guardar

Jugar

- Ingreso de los datos a la base



## Simulación

Tema: Inteligencia Artificial 1.

## Examen Final

```
def pulsar():
#aquí va neo4j
    aux = 0
    unique_transaction_results = execute_transactions(unique_transaction_types, True)
    with neo4j_driver.session() as session:
        if not unique_transaction_results:
            session.write_transaction(neo4j_crear_usuario, nombreU.get(), latitud.get(), longitud.get())
            session.write_transaction(neo4j_crear_ciudad, ciudad.get())
            session.write_transaction(neo4j_crear_relacion_ciudad_usuario, ciudad.get(), nombreU.get())
            session.write_transaction(neo4j_crear_relacion_pais_ciudad, "Ecuador", ciudad.get())
        else:
            for i in unique_transaction_results:
                if ciudad.get() == i:
                    aux = 1;
            if aux == 1:
                session.write_transaction(neo4j_crear_usuario, nombreU.get(), latitud.get(), longitud.get())
                session.write_transaction(neo4j_crear_relacion_ciudad_usuario, ciudad.get(), nombreU.get())
            else:
                session.write_transaction(neo4j_crear_usuario, nombreU.get(), latitud.get(), longitud.get())
                session.write_transaction(neo4j_crear_ciudad, ciudad.get())
                session.write_transaction(neo4j_crear_relacion_ciudad_usuario, ciudad.get(), nombreU.get())
                session.write_transaction(neo4j_crear_relacion_pais_ciudad, "Ecuador", ciudad.get())
```

5. Realizar el sistema con una interfaz gráfica para acceder y probar el sistema.



## Simulación

Tema: Inteligencia Artificial 1.

## Examen Final

tk

Ciudad:

Galapagos

Usuario:

SOLEDAD

Guardar

Latitud:

1238123

Jugar

longitud:

1432233

(West)

+-----+

| R: | : : G |

| : | : : |

| : : : : |

| : : : : |

| : : : : |

| : : : : |

| : : : : |

+-----+

(South)

+-----+

| R: | : : G |

| : | : : |

| : : : : |

| : : : : |

| : : : : |

| : : : : |

| : : : : |

+-----+

(South)

Score 9

Score Fínal: 7.92

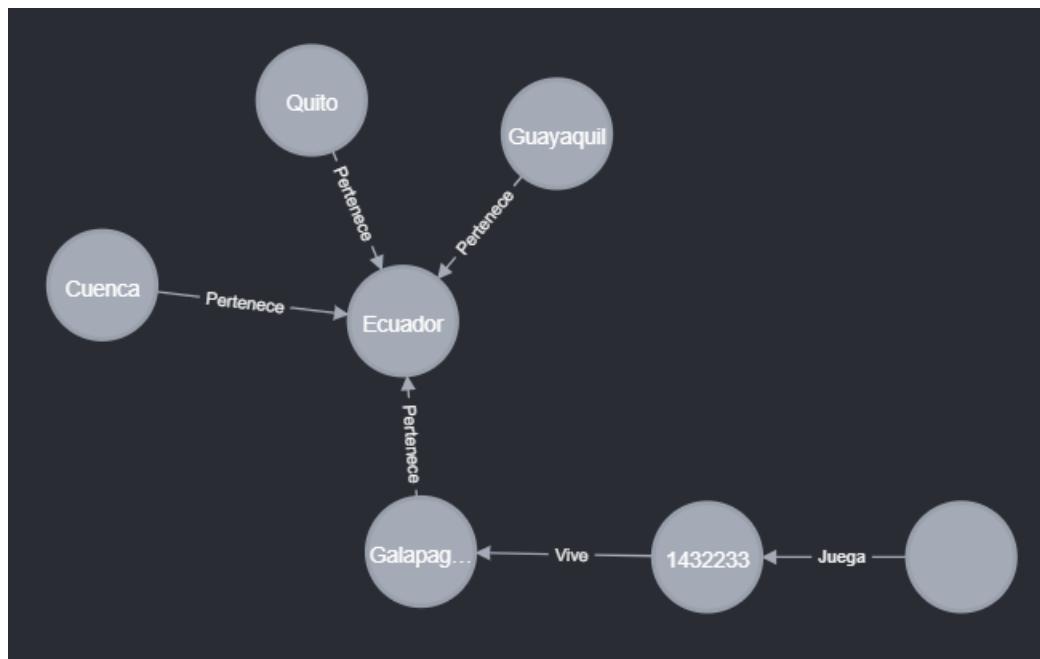




## Simulación

Tema: Inteligencia Artificial 1.

## Examen Final




- Prueba 2 Jugamos con la misma persona



## Simulación

Tema: Inteligencia Artificial 1.

## Examen Final

 tk — □ ×

Ciudad:

Usuario:

Latitud:

longitud:

Guardar

Jugar

(North)

+-----+

| R: | : | G |

| : | : |

| : : : |

| | : | : |

| Y | : | B: |

+-----+

(North)

+-----+

| R: | : : G |

| : | : : |

| : : : : |

| | : | : |

| Y | : | B: |

+-----+

(East)

Score 7

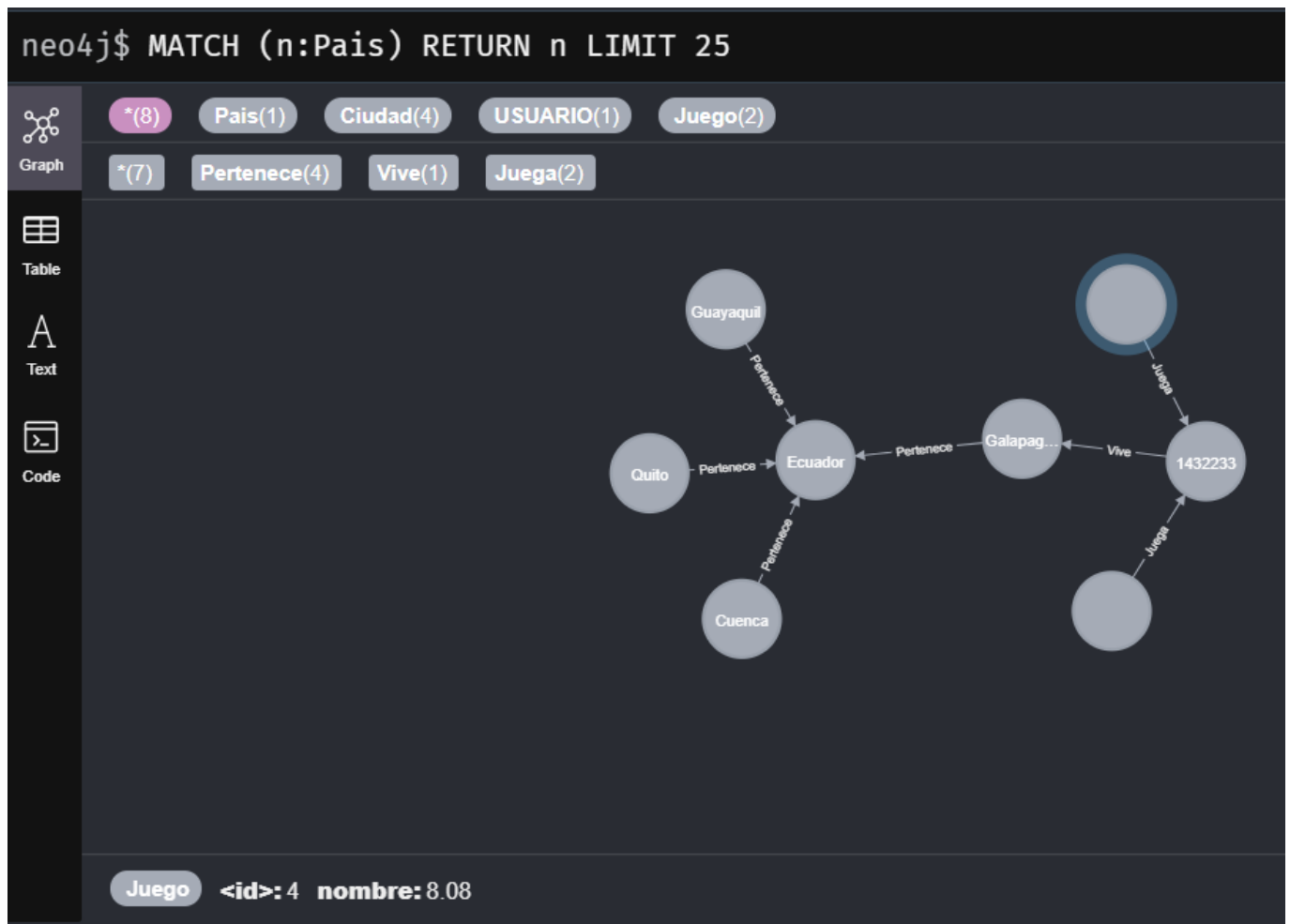
Score Fínal: 8.08



## Simulación

Tema: Inteligencia Artificial 1.

## Examen Final



- Prueba 3 diferente persona en diferente ciudad



## Simulación

Tema: Inteligencia Artificial 1.

### Examen Final

tk

Ciudad:

Usuario:

Latitud:

longitud:

Guardar

Jugar

(East)

+-----+

| R: | : : G |

| : | : : |

| : : : : |

| | : | : |

| Y | : | B: |

+-----+

(East)

+-----+

| R: | : : G |

| : | : : |

| : : : : |

| | : | : |

| Y | : | B: |

+-----+

(North)

Score 8

Score Fínal: 7.95



## Simulación

Tema: Inteligencia Artificial 1.

## Examen Final

