

# Hoja de Trabajo 1

En la industria no me sirve de mucho que sepan recitar definiciones de memoria si no pueden identificar cuando un entorno es estocástico o cuando su métrica de desempeño está mal planteada. Esta hoja de trabajo está diseñada para "aterrizar" los conceptos abstractos que vimos (Racionalidad, PEAS, Tipos de Entorno) a escenarios reales y de diseño de software. Recuerden justificar sus respuestas, se espera ver un análisis profundo, no respuestas de una línea.

## Instrucciones

- Esta es una actividad en grupos de no más de 2 integrantes.
  - Recuerden unirse al grupo de canvas
- No se permitirá ni se aceptará cualquier indicio de copia. De presentarse, se procederá según el reglamento correspondiente.
- Tendrán hasta el día indicado en Canvas.
  - No se confien, aprovechen el tiempo en clase para entender todos los ejercicios y avanzar lo más posible.

## Task 1 - Análisis de Racionalidad y Métricas

En la presentación definimos que un agente es racional si "para cada estado del ambiente, la función de transformación mapea acciones correctas" que maximizan el éxito. Sin embargo, el éxito depende de cómo lo definamos.

Imaginen que están diseñando un agente de IA para el sistema de semáforos de la Ciudad de Guatemala (zona de alto tráfico).

1. Propongan dos (2) métricas de desempeño distintas para este agente
  - a. Métrica A: Enfocada puramente en el flujo vehicular.  
Minimizar el tiempo que los semáforos tienen de espera a los vehículos
  - b. Métrica B: Enfocada en la seguridad del peatón.  
Garantizar que los peatones tengan un buen tiempo para cruzar completamente cada calle
2. Describa un escenario específico donde el agente actuaría de manera "racional" bajo la Métrica A, pero esa acción sería considerada un desastre bajo la Métrica B  
Un lugar en donde exista mucho tráfico vehicular, pero a la vez que pase mucho peatón, que cumpla la métrica A, es que, de bastante tiempo para cruzar a los vehículos, ejemplo, unos 5 minutos, en donde entra el desastre con la métrica B, es que, empezaría a impacientar a los peatones, y que estos mismos se crucen el rojo, haciendo que ocurran accidentes, además si el semáforo no les da el tiempo suficiente para poder cruzar.
3. Basándose en la diapositiva de Incertidumbre, explique por qué este entorno de tráfico nunca podrá ser Completamente Observable y cómo la "Limitante en sensores" afecta la racionalidad de su agente.  
Porque empezando con los sensores, no siempre se puede saber todo lo que los rodea, por ejemplo, tienen un límite de visión, de igual manera, no se sabe como sean los peatones, si van con prisa, o incluso, los sensores pueden no detectar a los demás peatones que irían a cruzar.

## Task 2 – PEAS y Entornos

Consideren y analicen el siguiente sistema propuesto:

Sistema: Un Robot Autónomo de Limpieza de Paneles Solares instalado en una granja solar en el desierto de Zacapa. El robot se mueve sobre rieles instalados en los paneles, tiene una cámara para detectar suciedad/daños, un brazo con cepillo y agua, y una conexión a internet para recibir reportes del clima. En base a esto responda lo siguiente

1. Complete la siguiente descripción del agente:
  - a. P (Performance/Desempeño): (Mencione al menos 3 indicadores cuantificables).
    - Eficacia al limpiar la suciedad
    - Consumo de agua
    - Tiempo en el que limpia los paneles
  - b. E (Environment/Entorno): (¿Qué rodea al agente y con qué interactúa?).
    - Una granja
    - Paneles que limpia
    - Suciedad y daños
  - c. A. (Actuators/Actuadores): (Mecanismos para afectar el entorno).
    - Brazo con cepillo de limpieza
    - Motors para moverse en el riel
    - Cámara para poder detectar lo que debe hacer
  - d. S (Sensors/Sensores): (Entradas de percepción). Asumiendo que no hace ningún preprocesamiento, ¿qué métrica usaría para medir el desempeño de su modelo?
    - Camara
    - La conexión a internet
    - Sensores de nivel de agua
2. Clasifique el entorno de este robot según las 4 dimensiones vistas en clase. Debe justificar cada elección (una elección sin justificación tiene valor de 0 pts).
  - a. ¿Completa o Parcialmente observable?
    - Podría ser parcialmente observable, ya que, a pesar de que cuenta con una cámara, se mueve solo dentro de unos rieles, por lo que, algunas veces no podría verse completamente toda la suciedad o daños que se encuentren más lejos.
  - b. ¿Determinístico o Estocástico? (Considere el factor clima y suciedad).
    - Estocástico, ya que, a pesar de que siempre siga haciendo mismo el robot, habrá días en los que no esté tan sucio, como días en los que se encuentre muy sucio y esto también se ve afectado por el clima
  - c. ¿Discreto o Continuo? (Analice los estados del movimiento y la suciedad).
    - Dicreto, ya que, su movimiento sería, si encuentra daño o suciedad, entonces procede a limpiar, por lo que hay soluciones finitas.
  - d. ¿Benigno o Adverso? (¿El polvo "juega" en contra del robot intencionalmente?)
    - Sería adverso, ya que, algunas veces el clima podría afectar que por ejemplo, el robot se encuentre limpiando y cambia el clima, por lo que debería de limpiar más, en este caso el clima y la suciedad sería lo que busca minimizar la métrica de desempeño

## Task 3 - Modelado

En las diapositivas discutimos el flujo Modeling -> Inference -> Learning. Asuma que queremos que nuestro robot de limpieza aprenda a optimizar el uso de agua. Para ello responda:

1. Describa abstractamente cómo representaría el "estado" de un panel solar en una estructura de datos (grafo, matriz, vector, etc.).

Se propone representar el estado de un panel solar mediante un vector que contenga las variables relevantes para la toma de decisiones:

estado\_panel = [nivel\_suciedad, posición\_en\_riel, temperatura\_panel, humedad\_ambiente, tiempo\_desde\_ultima\_limpieza]

Donde nivel\_suciedad es un valor entre 0-100 (0 = limpio, 100 = muy sucio), posición\_en\_riel indica la ubicación del panel, y las demás variables capturan las condiciones ambientales y de mantenimiento. Alternativamente, para modelar toda la granja solar podría utilizarse una **matriz** donde cada fila representa un panel, o un **grafo** donde los nodos son paneles y las aristas representan distancias en los rieles.

2. Si el robot tiene un modelo del panel, ¿qué significa "inferir" la siguiente acción? Relacionarlo con:

$$f(\text{estado}) \rightarrow \text{accion}$$

Inferir la siguiente acción significa que el modelo utiliza el estado actual del panel para predecir la acción óptima, siguiendo la función:

$$f(\text{estado}) \rightarrow \text{acción}$$

Por ejemplo, si nivel\_suciedad = 80 y humedad\_ambiente = 5%, el modelo infiere "limpiar con 2 litros de agua". Si nivel\_suciedad = 30 y humedad\_ambiente = 40%, infiere "limpiar con 0.5 litros". El modelo mapea cada estado observado a la acción que optimiza el uso de agua manteniendo la eficiencia de limpieza.

3. ¿Qué parámetro del modelo debería ajustarse si el robot nota que, tras limpiar un panel muy sucio con poca agua, el panel sigue sucio?

El parámetro a ajustar sería la cantidad de agua asignada para niveles altos de suciedad. Si el modelo tenía la regla “si nivel\_suciedad > 70 → usar 1 litro” y el panel quedó sucio, debe actualizarse a si “nivel\_suciedad > 70 → usar 2 litros.”

Este ajuste representa la fase de Learning del paradigma visto en clase: el robot detecta que su predicción fue subóptima y modifica los parámetros del modelo para mejorar decisiones futuras en situaciones similares.

## Task 4 – Práctica

No se requiere que resuelvan un algoritmo complejo de búsqueda. Se evaluará que puedan traducir la *Arquitectura de Agente* (Slide 26-29) a una estructura de Código Orientado a Objeto. Para esto debe escribir un script que implemente la estructura básica de un Agente Reflejo Simple para un entorno de "Termostato Inteligente".

- Debe crear una clase Environment que contenga:
  - Una variable para la temperatura actual (inicialícela aleatoriamente).
  - Un método get\_percept() que simule los sensores (devuelve la temperatura).
  - Un método update(action) que reciba la acción del agente y modifique la temperatura (ej: si la acción es "enfriar", bajar la temperatura).
- Debe crear una clase Agent que contenga:
  - Un método act(perception) que implemente la lógica  $f(\text{estado})$
  - Si temperatura > 25, retornar "enfriar".
  - Si temperatura < 18, retornar "calentar".
  - De lo contrario, retornar "esperar".
- Implemente un ciclo main (Ciclo de Percepción/Acción) que corra por 10 iteraciones, imprimiendo en cada paso:
  - El estado actual (Temperatura).
  - La acción elegida por el agente.
  - El nuevo estado después de la acción.

NOTA: No usen librerías de IA aún (no torch, no sklearn). Usen Python puro random y clases.

## Entregas en Canvas

1. Documento PDF con las respuestas a cada task
2. Archivo .py, o link a repositorio de GitHub (No se acepta entregas en otros medios)

## Evaluación

1. [1.5 pt] Task 1
2. [1.5 pt] Task 2
3. [1.0 pt] Task 3
4. [1.5 pt] Task 4

Total 5.5pts