

## Evaluación Técnica ICASS

1. Diseñé la mayor parte del modelo ER para la simulación, el cual consta de cuatro entidades principales: Hormiga, Colonia, Actividad y Feromona, además de tres relaciones clave. La primera relación conecta Colonia y Hormiga, con una cardinalidad 1: N; una Colonia puede tener múltiples hormigas, mientras que cada hormiga pertenece únicamente a una Colonia. La segunda relación, llamada Realiza, vincula Hormiga y Actividad con una cardinalidad N:M; esto permite que una hormiga (por ejemplo, de tipo soldado) pueda realizar múltiples actividades, y que una actividad pueda ser llevada a cabo por múltiples hormigas de diferentes tipos. La tercera relación, Emite, conecta Hormiga y Feromona, también con una cardinalidad N:M; esto significa que una hormiga puede emitir múltiples feromonas, y a su vez, una feromona puede ser emitida por varias hormigas diferentes (con distintos identificadores). Cada entidad posee atributos propios, los cuales fueron definidos en base a los requerimientos del problema planteado. Esta estructura fue diseñada para enfocarse en los elementos esenciales del sistema, como las Hormigas, las Actividades, las Colonias y la comunicación a través de Feromonas, mientras que algunos detalles adicionales, como la reserva de comida o los huevos, fueron omitidos en esta versión.
2. Las consultas se encuentran en el archivo de texto Pregunta 2.
3. **Array:** En un array, podemos almacenar los diferentes tipos de feromonas, tales como Amenaza, Defensa del hormiguero, Comida hallada, etc. Esta estructura es la más adecuada debido a que los tipos de feromonas son strings y conocemos de antemano su cantidad, la cual no varía.  
**Lista:** Las listas son estructuras de datos flexibles que pueden crecer según sea necesario. Por lo tanto, podemos utilizar una lista para almacenar los huevos que la hormiga reina deposita en una colonia.  
**Cola:** Utilizamos una cola, siguiendo el patrón FIFO (First In, First Out), para almacenar a los príncipes, ya que después de cumplir su función, deben ser expulsados de la colonia. Esta elección resulta eficiente, ya que la adición y eliminación de príncipes se realiza en tiempo constante (O(1)).  
**Diccionario:** Empleamos un diccionario para mapear los diferentes tipos de comida con las reservas disponibles en el hormiguero. Las claves representan los tipos de comida (por ejemplo, de obreras, de soldados, de nodrizas, etc.), mientras que los valores corresponden a las cantidades disponibles de cada tipo de comida.

4. Plan de Desarrollo del Simulador de Hormigueros:

**Metodología Ágil:** Scrum.

**Equipo:** 2 desarrolladores y 1 tester.

**Sprint 1: Planificación y Diseño**

Duración: 2 semanas

Actividades:

- Estudiar problema a resolver.
- Llevar a cabo una reunión inicial para definir requisitos funcionales y no funcionales.
- Diseñar el modelo ER.
- Crear el backlog del producto.

**Sprint 2: Implementación de la Lógica**

Duración: 3 semanas

Actividades:

- Implementar las clases y métodos para simular el comportamiento de las hormigas.
- Implementar la comunicación mediante feromonas.
- Realizar pruebas unitarias y de integración.

**Sprint 3: Correcciones y Pruebas Finales**

Duración: 2 semanas

Actividades:

- Corregir errores y optimizar el código.
- Realizar pruebas finales, incluyendo usabilidad.

5. Una actividad del sprint 2 es la creación de las clases para representar cada tipo de hormiga en el simulador. En el archivo "codigo\_base.py", se muestra un código en Python que representa un avance en el desarrollo del simulador.