

Debes programar un sistema que gestione las tareas de reparación de una granja de servidores. El sistema debe cumplir con los siguientes requisitos:

1. **Priorización:** Las tareas se procesan según su nivel de urgencia (1 = Crítico, 5 = Informativo) y tipo (hardware, software).
2. **Unicidad:** No se pueden añadir tareas con la misma descripción y tipo si ya están pendientes (usa un `set` de tuplas (tipo, descripción) para controlarlo).
3. **Registro histórico:** Debes mantener un historial de cuántas tareas de cada prioridad se han completado (usa un `dict`).

Se debe implementar la clase “GestorIncidencias” con las estructuras de datos y métodos necesarios:

- **Atributos:**
 - o Una lista independiente para cada tipo de incidencia (software, hardware) con la que implementar el heap de cada cola.
 - o Conjunto para evitar duplicados (tipo, descripción)
 - o Diccionario de histórico
- **Métodos**
 - o `agregar_incidencia(prioridad, tipo, descripción)`: Inserta la tarea en su correspondiente cola si la descripción y tipo no está duplicada.
 - o `atender_siguiente(cola)`: Extrae y muestra la tarea más urgente.
 - o `ver_proxima(cola)`: Muestra la tarea más urgente sin extraerla del sistema.
 - o `resumen_estadistico()`: Muestra el conteo de tareas completadas por prioridad.

Python compara las tuplas elemento a elemento. Si se empata en el primer elemento, se compara el segundo.