

FINAL DE PARADIGMAS DE PROGRAMACION

Se construye un modelo con estas clases:

```
Juzgado
mi: derivarAExperto: expediente

Perito
mi: notificarConsulta: expediente

Facultad
mi: notificarConsulta: expediente
mi: puedeOpinarSobre: expediente
```

Las siguientes clases heredan de Perito: Contador, IngSistemas, Veterinario con la siguiente implementación en la clase Juzgado

```
derivarAExperto: expediente
    (self expertos detect: [:e | e puedeOpinarSobre: expediente])
    notificarConsulta: expediente
```

(detect: devuelve un elemento cualquiera de una colección que cumpla con la condición).

Se pide:

- Indicar en qué clase/s falta implementar el método puedeOpinarSobre: .
- Describir el uso de polimorfismo que se aprecia en este ejemplo, indicando en particular quién lo está aprovechando, o sea, qué código sería más complejo si no hubiera polimorfismo, y en qué consistiría esta complejidad.
- Nos dicen que para notificar a los ingenieros en sistemas, además de lo que se hace para cualquier perito, hay que enviarle un mail. Indique cómo resolvería este agregado, y qué concepto/s asociados al paradigma de objetos está usando.
- Usando como ejemplo las clases incluidas en este ejemplo, indique qué criterio usaría para decidir, dado un conjunto de clases cuyas instancias deben ser todas polimórficas entre sí, en qué casos usar la herencia y en qué casos no.

Punto 4

En la materia decimos que en los paradigmas lógico y funcional se respeta la transparencia referencial, mientras que en el de objetos no necesariamente.

Indique qué quiere decir esto en términos de cada paradigma

En otras palabras, ¿cómo terminarías la frase que empieza así

"en el paradigma lógico siempre hay transparencia referencial, entonces ..."?

Lo mismo para los otros dos paradigmas.

En el caso específico de objetos, dar un ejemplo que colabore con la explicación.