

NOMBRE:

(no olvide firmar y numerar cada hoja: "HOJA 2 DE X")

--

1) Se quiere resolver el siguiente requerimiento:

“Un doctor lleva la historia clínica de los pacientes que atiende. Por cada visita (atención a un paciente) se registra la fecha y medicamentos que le recetó (dosis y nombre de la medicación genérica). Se quiere saber la cantidad de visitas en las que al paciente le dieron un determinado medicamento”

Solución propuesta:

>>Paciente

```

cantidadDeVisitasEnLasQueLeDieron: unMedicamento
| visitasAux |
visitasAux := 0.
visitas do: [ :visita | visita dosisMedicamentos do:
              [ :dosis | dosis medicamento codigo =
                  unMedicamento codigo
                  ifTrue: [ visitasAux := visitasAux + 1 ]
              ] ].
^visitasAux

```

- Si objetos modela la realidad, ¿el código anterior no debería estar en la clase Doctor? Justifique su respuesta.
- Realice los cambios que crea necesario teniendo en cuenta el concepto **delegación** (que cada objeto haga lo que le corresponde). Justifique su solución.
- ¿Qué ventajas le encuentra la solución en b) por sobre la solución existente? Justifique a través de un ejemplo concreto (ej. con un requerimiento particular)

Se quiere resolver el mismo problema en Lógico. Se propone la siguiente solución:

```

visita(gutierrez, fecha(10, 11, 2009), [dosis(ranitidina, 200)]).
visita(gutierrez, fecha(16, 11, 2009), [dosis(trimebutina, 400),
                                         dosis(ranitidina, 100)]).

vecesQuePidio(Paciente, Medicamento, Cantidad):-
    findall(Paciente, (visita(Paciente, _, Medicamentos),
                      member(dosis(Medicamento, _), Medicamentos)),
            Lista),
    length(Lista, Cantidad).

```

- Cambie el código del predicado `vecesQuePidio/3` utilizando predicados auxiliares. ¿Qué ventaja encuentra a este cambio? Justifique a través de un ejemplo concreto (ej. con un requerimiento particular)
- El predicado original `vecesQuePidio/3` no es inversible. Justifique por qué no lo es, qué cambiaría en el código para que sí lo sea.
- ¿Qué ventaja le da la solución inversible en Lógico por sobre la solución en Objetos? Justifique su decisión.
- ¿De qué otra manera podría haber definido la información de los predicados `visita`? Codifique la solución indicando qué conceptos aparecen y por qué.

2) Para la misma aplicación tenemos el siguiente requerimiento:

“Se quiere conocer la cantidad de medicación que tuvo un paciente para una droga específica”

Funcional

```
visitasGutierrez = [(10, 11, 2009, "ranitidina", 200),
                    (16, 11, 2009, "trimebutina", 400),
                    (16, 11, 2009, "ranitidina", 100)]
```

```
medicamento (_, _, _, m, _) = m
```

```
cantidad (_, _, _, _, c) = c
```

```
cantidadDeMedicacion droga visitas
    = (sum . map (cantidadDe droga)) visitas
```

```
cantidadDe droga dosis
    | ((==) droga . medicamento) dosis = cantidad dosis
    | otherwise                          = 0
```

- ¿Por qué cree ud. que se representan las visitas como una lista? ¿Podría intentar utilizar un formato similar a la solución planteada para el paradigma Lógico? Justifique su respuesta.
- Explique dónde aparece el concepto de aplicación parcial, justifique su decisión.
- Relacione en qué medida la aplicación parcial favorece/desfavorece evitar la repetición de código. Justifique su respuesta.
- Explique dónde aparece el concepto función de orden superior, y qué ventaja supone su utilización en la solución propuesta.

Otro programador acerca una propuesta diferente:

```
cantidadDeMedicacion droga (visita:visitas)
    | medicamento visita == droga = cantidad visita +
                                     cantidadDeMedicacion droga visitas
    | otherwise                      = cantidadDeMedicacion droga visitas
```

- ¿Cuál es el error que tiene la solución? Justifique la respuesta. (La solución tiene un error / bug porque no funciona, más allá de que compile. No hay “error conceptual”)
- “Las soluciones con mayor grado de declaratividad suelen tener menos errores propios de la construcción de un algoritmo”. Justifique a favor o en contra de la frase anterior, teniendo en cuenta el punto f).