PARADIGMAS DE PROGRAMACIÓN – EXAMEN FINAL 26/09/2007

LEGAJO: APELLIDO Y NOMBRE:

1 - Dadas las siguientes definiciones dadas en los lenguajes Haskell y Prolog

i) la función **concatenar**

```
concatenar [] ys = ys
concatenar (x:xs) ys = x:concatenar xs ys
```

ii) el predicado **concatenar/3**, definido por las siguientes cláusulas

```
concatenar([], Ys, Ys).
concatenar([X|Xs], Ys, [X|Zs]):-concatenar(Xs, Ys, Zs).
```

a) Describa cómo se aplican, en cada item, los conceptos de

- inversibilidad
- pattern matching
- tipos de datos genéricos/polimorfismo

b) ¿Qué puede decir acerca de las consultas que se pueden hacer en cada caso? Relacione con concepto/s visto/s en la materia.

2 - Se presenta el siguiente requerimiento:

"Dado un conjunto de personas, filtrar aquellas que son viejas. Decimos que una persona es vieja cuando supera los 75 años".

a) Plantear un modelo de objetos que permita resolver el requerimiento, de manera de no plantear toda la solución como un script en un workspace, sino que pueda resolverse a través del envío de un mensaje con la colección como parámetro, y que dicho mensaje retorne la colección filtrada. Es decir:

```
coleccionFiltrada := objetoReceptor mensaje: coleccionDePersonas
```

b) Amplíe la resolución del punto a) de manera de soportar el siguiente requerimiento:

"Además de la posibilidad de filtrar las personas viejas, el sistema deberá ser capaz de brindar la siguiente funcionalidad:

- *de una colección de casas, filtrar aquéllas que son viejas (construidas antes del año 1940)*
- *de una colección de autos, filtrar aquellos que son viejos (cuya patente posee un indicador de la provicina de origen, no nulo). "*

¿Qué cosas tuvo que cambiar en la solución?¿Encuentra ventajas en el planteo hecho en objetos? Enumérelas.

c) Plantear una solución en paradigma funcional que permita resolver el punto a). Defina funciones (o utilice funciones existentes en el prelude), e invóquelas. Respete la idea general que planteó en la solución previa.

3- Dada la clase EmpresaDeTransporte, se programa el siguiente método:

```
cargaMaxima
   "Retorna la carga máxima de la flota de transportes del receptor"
   |transportesDisponibles carga|
   carga := 0.

   transportesDisponibles := transportes select: [:transp |
      (transp viajeActual = nil) & (transp tallerDondeEsta = nil)].

   transportesDisponibles do: [:transp |
      transp acoplados do: [:acopl |
         carga := carga + acopl cargaMaxima
      ]
    ].
```

PARADIGMAS DE PROGRAMACIÓN – EXAMEN FINAL 26/09/2007

LEGAJO: APELLIDO Y NOMBRE:

```
^carga
```

Suponga que todos los mensajes enviados son comprendidos por sus receptores

a) Identifique **todos** los objetos que intervienen en el cálculo de la carga máxima de una instancia de la clase EmpresaDeTransporte.
b) La solución planteada, claramente no está haciendo uso de un concepto del paradigma de objetos (el cual permitiría reducir la cantidad de código del método). ¿Cuál es dicho concepto? Señale la/s porcion/es de código donde nota su ausencia.
c) Modifique la solución para llegar a una implementación que funcione igual y en la que sí se aproveche el concepto identificado en el ítem anterior.