

FINAL DE PARADIGMAS DE PROGRAMACION

Punto 1

En el sistema de facturación de una compañía telefónica aparece este código:

```
FacturadorTelefonico
  costoLlamada: unaLlamada para: unCliente
  | costoPorMinuto costoBasico esClienteParticular porcDescuentoCliente costo |
  costoPorMinuto := 0.12.
  costoBasico := costoPorMinuto * unaLlamada duracion.
  (unCliente esParticular)
    ifTrue: [porcDescuentoCliente := 10]
    ifFalse: [porcDescuentoCliente := 5].
  costo := costoBasico * (1 - (porcDescuentoCliente / 100)).
^costo

#ClienteParticular
esParticular
^true

#ClienteComun
esParticular
^false
```

El costo básico de una llamada es el costo antes de aplicar los descuentos correspondientes.

El método **duración**, que calcula la duración de una llamada, ya está hecho.

La diferencia entre los clientes particulares y los otros es sólo el porcentaje de descuento que se aplica a sus llamadas.

- a) Identificar y justificar conceptualmente los problemas de esta solución
- b) Hacer una nueva solución correcta, mostrando los cambios
- c) Modificar la solución anterior para contemplar los siguientes requerimientos.
 - c.1) Todos los clientes pueden hacer llamadas de larga distancia, que tienen un costo básico del doble que las locales por minuto, con un mínimo de \$2 por llamada, mientras que el precio de las llamadas locales se sigue calculando de igual manera.
 - c.2) Hay una nueva categoría de clientes, llamada clientes sociales, a quienes se les aplica el 25% de descuento en todas sus llamadas.
- d) ¿Qué nuevos conceptos se aplican en la solución? Explicar las ventajas de haberlos utilizado.

Punto 2

- a) Basándose en el paradigma lógico, compare los conceptos de listas y funtores, explicando diferencias y semejanzas.
- b) Ahora pensemos en las listas de Prolog y de Haskell. Tengo en Prolog, el predicado sumaLista/2 definido así:

```
sumaLista([], 0).
sumaLista([X|Xs], N):- sumaLista(Xs,N1), N is N1 + X.
```

en Haskell, la función sumaLista/1 definida así:

```
sumaLista [] = 0
sumaLista (x:xs) = x + sumaLista xs
```

- b.1) ¿Qué diferencias hay en el comportamiento del Swi Prolog y del Hugs, respectivamente, ante estas

FINAL DE PARADIGMAS DE PROGRAMACION

consultas?

- ?- sumaLista(["alfa"],X).
- > sumaLista ["alfa"].

b.2) Justifique indicando con qué concepto está relacionada esta diferencia. En particular ¿se debe a una diferencia entre los paradigmas lógico y funcional?

Punto 3

Una fábrica de autos quiere informatizar parte de su operación. En particular:

- a) el análisis de qué combinaciones puede armar para cada modelo a partir de las distintas versiones de motores, accesorios y colores disponibles, y de qué combinaciones en particular conviene ofrecer en distintos mercados, a partir de información sobre los gustos de cada mercado.
Dada una combinación (p.ej. Ford Ka motor 1.6 verde con aire acondicionado) se requiere saber para qué mercados puede funcionar (p.ej. para Zaire porque en Zaire andan bien los autos chicos con aire acondicionado), y viceversa, se requiere poder averiguar qué combinaciones pueden funcionar para Zaire.
- b) el control del stock de partes y productos fabricados, a partir del registro de las compras y los partes de fabricación.

Entre los tres paradigmas estudiados en la materia, ¿cuál elegiría para cada una de estas partes? Justifique su respuesta.
