

Notas generales

- Para aprobar es necesario simultáneamente:
 - obtener 8 puntos de 14, y
 - obtener al menos la mitad de los puntos en cada paradigma.
- Las preguntas choice o V/F:
 - no serán consideradas si no están justificadas, y
 - se justifican mediante explicaciones y/o código a criterio del alumno.
- Y recordá: en todas tus respuestas sé puntual, no pierdas el foco de lo que se pregunta. Respuestas en exceso generales son tan malas como respuestas incompletas.



Punto A (6 puntos)

En una agencia de publicidad gráficas existen dos tipos de publicidades:

- Publicidades "comunes": se publican en diarios y revistas de la agencia, y tienen un costo para el cliente que varía según el plan que contrate:
 - Estandar: \$1000.
 - Platinum: \$800 si el cliente tiene activas más de 3 publicidades. Si no, \$1100.
 - Gold: \$1200 / cantidad de publicidades activas del cliente.
- Publicidades no tradicionales: se publican en distintos medios a elección, cada uno de los cuales tiene un costo propio. El costo de la publicidad no tradicional es la suma de estos costos.

Nota: no nos interesa modelar cómo el cliente contrata un plan o cuanto le sale éste; sólo estamos interesados en saber cuánto le cuesta una publicidad dada.

Tenemos la siguiente solución (se omiten los *getters* y *setters*):

```
#Cliente (vi: publicidades)

#PublicidadNoTradicional (vi: medios, estaActiva)
>> costo
  ^medios sum: [ :medio | medio costo ]

#Publicidad(vi: plan) (subclase de PublicidadNoTradicional)
>> costo: cliente
  (plan = 'estandar') ifTrue: [
    ^1000
  ].
  (plan = 'platinum') ifTrue: [
    ^((cliente publicidades select:[:publicidad | publicidad estaActiva]) size > 3)
      ifTrue: [800] ifFalse: [1100]
  ].
  (plan = 'gold') ifTrue: [
    ^1200 / ((cliente publicidades select: [ :publicidad | publicidad estaActiva ]) size)
  ].
```

1. Responder V/F, justificando adecuadamente: (0,5 puntos cada una)
 - a. Las instancias de PublicidadNoTradicional y Publicidad permiten su uso polimórfico para un tercero dado que todas entienden el mensaje costo:.
 - b. Para poder lograr el polimorfismo entre Publicidad y PublicidadNoTradicional, es necesario el uso de herencia.
 - c. Tiene sentido que Publicidad sea subclase de PublicidadNoTradicional, lo que permite además un buen uso de redefinición.
 - d. La solución presenta lógica repetida.

- e. En la clase Publicidad al modelar el plan como un string, se aprovecha el polimorfismo entre los planes.
 - f. La solución rompe el encapsulamiento del cliente.
2. Codificar una solución que resuelva los problemas identificados anteriormente (3 puntos).

Punto B (4 puntos)

1. Tenemos la siguiente definición de la función \circ :

$$f \ g \ x \ y = f \ (g \ x) \ (g \ y)$$
 - a. Dar su tipo más general (1 punto)
 - b. Dar un ejemplo de aplicación (que funcione adecuadamente) de la misma (1 punto)
2. Dada la siguiente definición de la función m :

$$m \ f \ g \ x = any \ f \ . \ map \ (g \ x)$$
 Para cada una de las siguientes consultas (0,5 puntos cada una)
 - a. $m \ (>5) \ (*) \ 2 \ [1, \ 2, \ 3]$
 - b. $m \ (+10) \ (+) \ 1 \ ['h', \ 'o', \ 'l', \ 'a']$
 - c. $m \ (>0) \ (+) \ 1 \ [1, \ 2.. \]$
 - d. $m \ (<0) \ (+) \ 1 \ [1, \ 2.. \]$
 Indicar y **justificar** si:
 - Se evalúa y termina adecuadamente, o bien
 - se evalúa pero no termina, o bien
 - no se evalúa debido a un error de tipos.

Punto C (4 puntos)

Dado el siguiente código:

```
todosLosVotantesHabilitados(Lista5):-
    todosLosAlumnosRegulares(Lista1),
    findall(D, esDocente(D), Lista2),
    append(Lista1, Lista2, Lista3),
    findall(G, esGraduado(G), Lista4),
    append(Lista3, Lista4, Lista5).
```

```
todosLosAlumnosRegulares(Alumnos):-
    findall(Alumno, esRegular(Alumno), Alumnos).
```

```
esRegular(alumno(_, _, FR)):- -- FR son los finales rendidos
    finalesRendidosUltimoAnio(FR, FRUA),
    length(FRUA, N),
    N >= 2.
```

Asumiendo que los predicados `finalesRendidosUltimoAnio`, `esGraduado` y `esDocente` funcionan adecuadamente y son completamente inversibles, indicar V/F **justificando**: (0,5 puntos cada una)

1. `todosLosAlumnosRegulares/1` es un predicado inversible.
2. La solución es poco expresiva.
3. La consulta `?- todosLosAlumnosRegulares(Alumnos).` tiene múltiples soluciones.
4. El uso de `length` es incorrecto, dado que tiene aridad 1.
5. La consulta `?- esRegular(ernesto).` da error.
6. `todosLosVotantesHabilitados` es un predicado inversible.
7. `todosLosVotantesHabilitados` aprovecha el polimorfismo entre docentes, graduados y alumnos regulares.
8. La solución podría ser reescrita y simplificada sin usar el predicado `append/3`.