

24-02-24 resolucion final

Parte A

1. a) V. Se repite logica en los if al consultar el tipo de plan
[Corrección: "No menciona lo que se puede generalizar entre publicidades"]
b) F. La responsabilidad de ese cálculo debería ser de Cliente, quien tiene las publicidades que conoce.
c) F. Los mensajes son distintos. Uno es costo() y el otro costo(cliente).
d) F. Para esta solución habría que agregar código en PublicidadTradicional.
- 2.

```
class Cliente {
    var property publicidades = []

    method cantPublicidadesActivas() {
        return self.publicidades().count({p => p.estaActiva()})
    }
    method costoTotal() {
        return self.publicidades().sum({p => p.calcularCosto(self)})
    }
}

class abstract Publicidad {
    var property estaActiva = true

    method calcularCosto(cliente) {
        if(self().estaActiva) {
            return self.costo(cliente)
        }
        else return 0
    }

    method costo(cliente)
}

class PublicidadNoTradicional extends Publicidad {
    var property medios = []

    method costo(cliente) {
        return self.medios().sum({m => m.tarifa()})
    }
}

class Medios { // De este no tuve correccion pero tampoco esta mal
    var property tarifa
}

class PublicidadTradicional extends Publicidad {
```

```

    var property plan

    method costo(cliente) {
        return plan.costo(cliente.cantPublicidadesActivas())
    }
}

interface Plan {
    method costo(cantActivas)
}

class PlanEstandar implementes Plan {
    method costo(cantActivas) {
        return 1000
    }
}

class PlanPlatinum implementes Plan {
    method costo(cantActivas) {
        if(3 < cantActivas) return 800
        else return 1100
    }
}

class PlanGold implementes Plan {
    method costo(cantActivas) {
        return 1200 / cantActivas
    }
}

```

Parte B

1. No. Esta solución valida que un alumno haya aprobado una de las correlativas de la materia que quiere rendir, no todas. Por ej, dado la siguiente base de conocimientos:

```

aproboCursada(ana, pdp, 2024/01/01).
correlativa(discreta, pdp).
correlativa(algoritmos, pdp).
aproboFinal(ana, discreta, 2022/01/01).

```

La consulta `puedeAnotarseAFinal(ana, pdp, _)` daría verdadero ya que existe una correlativa aprobada para ana (discreta).

2. No es inversible para el tercer parámetro ya que `añosLectivosTranscurridos` no lo es para su tercer parámetro. No sería posible hacerlo inversible porque no se puede acotar las fechas de final posibles para realizar el cálculo de los años lectivos.
- 3.

```

debeFinal(Alumno, Materia, FechaFirma) :-
    aproboCursada(Alumno, Materia, FechaFirma),
    not(aproboFinal(Alumno, Materia, _)).

```

```

puedeAnotarseAFinal(Alumno, Materia, Fecha) :-
    debeFinal(Alumno, Materia, FechaFirma),
    añosLectivosTranscurridos(FechaFirma, Firma, 0).
    // Correccion: aca me habia faltado el 0

puedeAnotarseAFinal(Alumno, Materia, Fecha) :-
    debeFinal(Alumno, Materia, _),
    forall( correlativa(Correlativa, Materia),
            aproboFinal(Alumno, Correlativa, _))

```

Parte C

1. $f :: \text{Num } a \Rightarrow (a \rightarrow \text{Bool}) \rightarrow [(b, [[a]])] \rightarrow \text{Bool}$
 [Corrección: Me tacharon el " $\text{Eq } a, b \Rightarrow$ " que puse originalmente y quedo así. No sé si habría que ponerle un tipo a 'b' o no. El resto ok]
2. a) V. Recibe como parámetro otra función (h)
 b) V. [Corrección: Mal justificado]
 c) [Corrección: todo mal jaja. Puse F y justifiqué pero está todo tachado así que supongo que es V]
 d) F. sumatoriaMayorA15 expresa mejor en lenguaje común lo que hace.
 e) V. [Corrección: Mal justificado]