

# Programación III

## Clase XI

*Andrés Fortier*

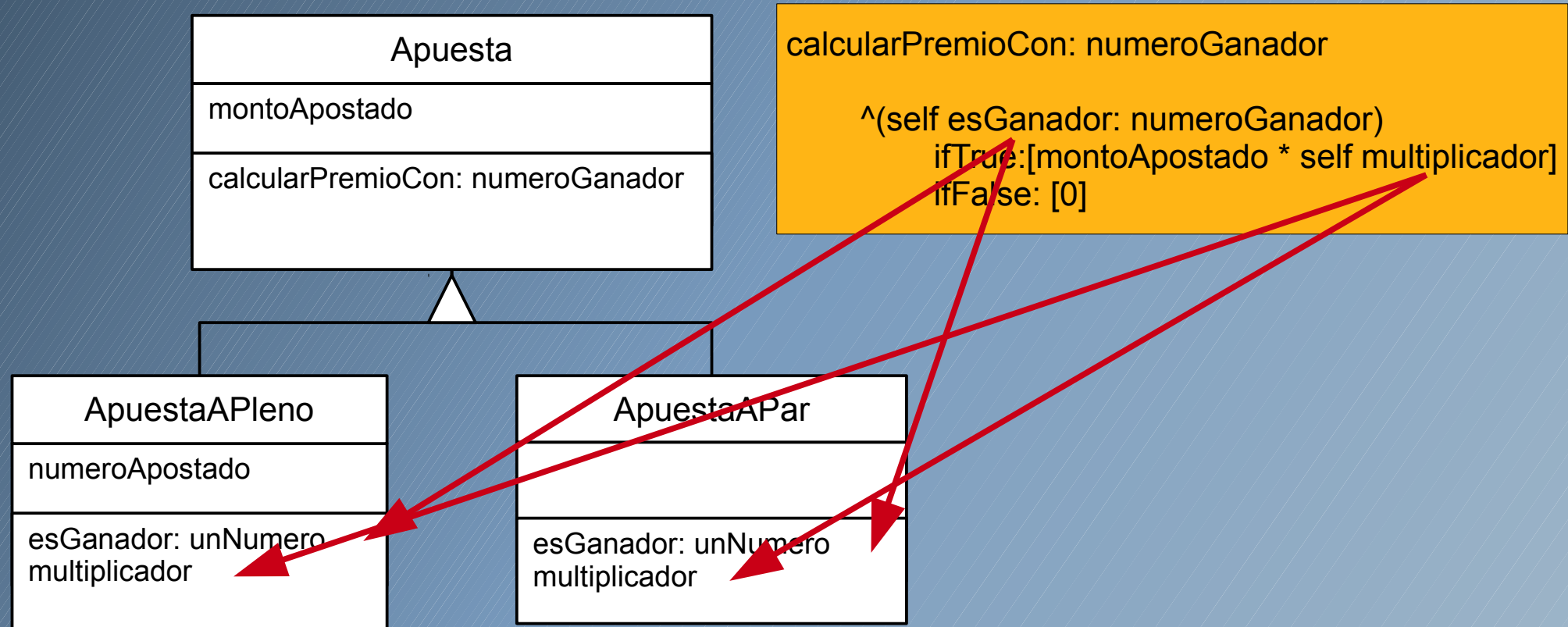
# Importante

- Pasamos el parcial del lunes de la semana que viene al jueves por la clase que no tuvimos la semana pasada.
- El sábado hay consulta.

# Volvamos a la ruleta

- Tenían que terminar el ejercicio.

# Ejercicio Ruleta





# Jerarquías de clases

- Para crear un tipo nuevo de apuesta debemos subclasificar Apuesta.
- ¿Cómo indicamos que todas las subclases deben implementar los mensajes `#esGanador:` y `#multiplicador?`.
- ¿Cómo indicamos que Apuesta todavía no tiene todo el comportamiento necesario y que, por ende, no se pueden crear instancias de la misma?

# Mensajes abstractos

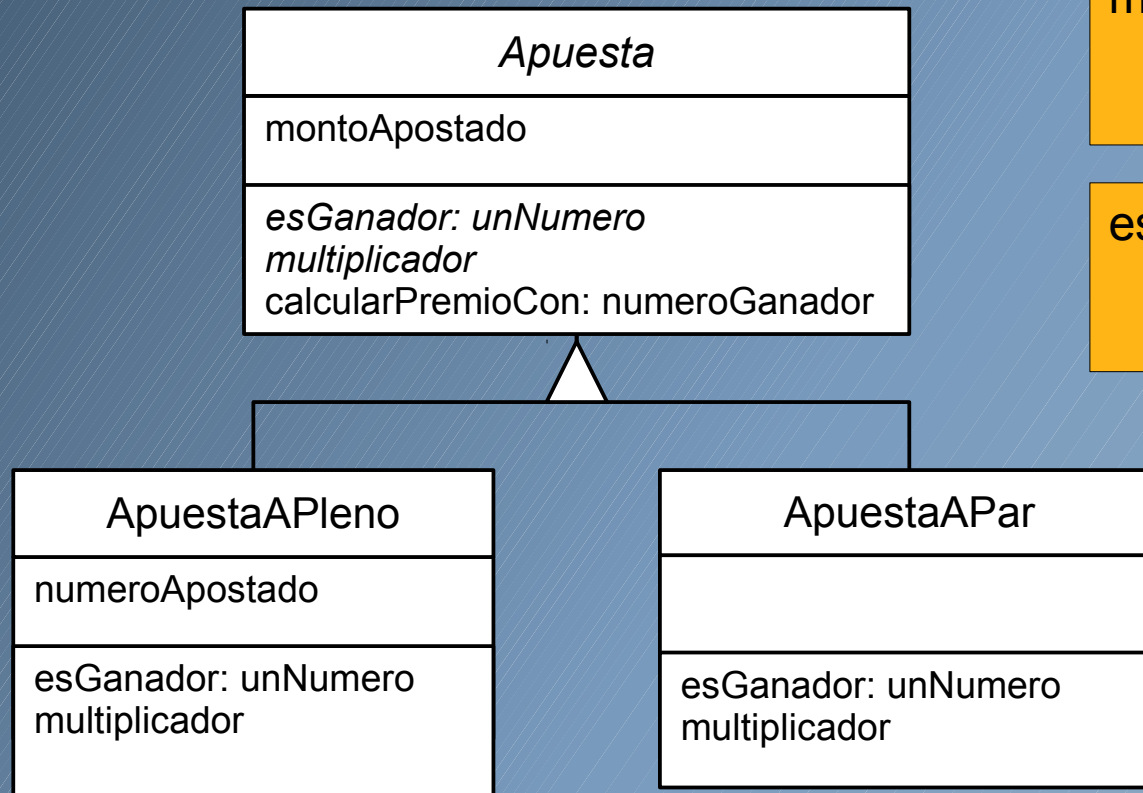
- Indican que una clase debería entender un determinado mensaje, pero no posee la información suficiente para implementarlo.
- Las subclases deben proveer una implementación.
- En UML se denotan con la letra cursiva.
- En Smalltalk se implementan utilizando el mensaje `#subclassResponsibility`.

# Clases abstractas

- Son aquellas que tienen al menos un mensaje abstracto.
- En UML se denotan con la letra cursiva.
- No hay forma de indicar esto en Smalltalk.



# Diagrama de clases



calcularPremioCon: numeroGanador

```
^(self esGanador: numeroGanador)
  ifTrue:[montoApostado * self multiplicador]
  ifFalse: [0]
```

multiplicador

^self subclassResponsibility

esGanador: unNumero

^self subclassResponsibility