



logofiuba.jpg

Trabajo Práctico 1 — Smalltalk

[7507/9502] Algoritmos y Programación III
Primer cuatrimestre de 2025

Alumno:	Iñaki Vydra
Número de padrón:	111505
Email:	ivydra@fi.uba.ar

Índice

1. Introducción

El presente informe reúne la documentación de la solución del primer trabajo práctico de la materia Algoritmos y Programación III que consiste en desarrollar un modelo de predicción de valores de criptomonedas según un divulgadores en Pharo 11 utilizando los conceptos del paradigma de la orientación a objetos vistos hasta ahora en el curso.

2. Detalles de implementación

En esta sección voy a mostrar dos partes del código donde se implementa polimorfismo para resolver problemas en vez de usar estructuras condicionales.

2.1. Polimorfismo en la clase Divulgador

Un Divulgador es un objeto que se encarga de hacer predicciones sobre alguna criptomoneda que conozca. Las predicciones se calculan aplicando una función a los atributos de `edad` y `patrimonio` del divulgador. Sin embargo, esta función varía según el estado del `ColorDePelo`, y al resultado se le aplica un ajuste que depende de la `Nacionalidad`.

Por ejemplo:

- Si el divulgador es `colorado` y `argentino`, el resultado es: $(\text{patrimonio}/10 + 3 \times \text{edad}) \times 0,75$
- Si es `morocho` y `extranjero`, sería: $((\text{edad} \times \text{patrimonio})/10\,000\,000) \times 1,1$
- La forma general es: `función(edad, patrimonio) × factorDeConfianza`

Este comportamiento lo implementé utilizando polimorfismo. Para ello, cree las clases `Colorado`, `Morocho` y `Rubio`, que entienden el mensaje `calcularValorCon: unaEdad yCon: unPatrimonio` para calcular el valor base de la predicción. Además, las clases `Argentina` y `Extranjero` entienden el mensaje `aplicarFactorDeConfianza: unValor` para aplicar el ajuste correspondiente.

```
valorPrediccion := colorDePelo calcularValorCon: edad yCon: patrimonio.  
valorPrediccion := nacionalidad aplicarFactorDeConfianza: valorPrediccion.
```

De esta manera, el Divulgador, al calcular el valor de una nueva predicción, delega el cálculo a sus atributos `colorDePelo` y `nacionalidad`, aplicando polimorfismo en ambos casos.

2.2. Polimorfismo en el criterio de búsqueda

La clase `AlgoPrediccionCripto` tiene un atributo `criterio` que se utiliza para buscar uno de sus divulgadores y devolver el que mejor se ajuste a dicho criterio.

Por ejemplo:

- Si el criterio es `Maximo`, se devuelve el divulgador que tenga la predicción más grande.
- Si el criterio es `Minimo`, se devuelve el divulgador que tenga la predicción más pequeña.

Para devolver el divulgador correcto, se implementó un algoritmo que itera sobre todos los divulgadores registrados y les solicita su mejor predicción según un criterio dado. Para responder a esta solicitud, cada divulgador debe iterar sus criptomonedas conocidas y aplicar ese criterio para determinar cuál usar para la predicción.

Esto implica que el criterio debe permitir comparar los valores de diferentes criptomonedas, así como también comparar predicciones de distintos divulgadores.

Para lograr esto, utilicé polimorfismo: creé las clases `Maximo` y `Minimo`, que entienden los mensajes `compararUnValor: unValor conOtro: otroValor` y agregue un método a la clase `Prediccion` que le permite compararse con otras predicciones en base a un criterio. De esta manera, las

Predicciones puede delegar en su atributo `criterio` el comportamiento de elegir entre una predicción u otra.

```
divulgadorSegunCriterio := divulgadores associations reduce: [
:mejorDivulgador :divulgadorActual |
| precisionActual |
mejorPrediccion ifNil: [
mejorPrediccion := (mejorDivulgador value) obtenerMejorPrediccionSegun: criterio.
].
precisionActual := (divulgadorActual value) obtenerMejorPrediccionSegun: criterio.

(mejorPrediccion compararConOtraPrediccion: precisionActual segunUnCriterio: criterio)
ifTrue: [
mejorDivulgador
]
ifFalse: [
mejorPrediccion := precisionActual.
divulgadorActual
]
].
```

De forma análoga, cada divulgador puede delegar en el criterio que recibe como parámetro el comportamiento de elegir una criptomoneda sobre otra:

```
criptomonedaSegunCriterio := criptomonedas associations reduce: [
:mejorCriptomoneda :criptomonedaActual |
(criterio compararUnValor: mejorCriptomoneda value conOtro: criptomonedaActual value)
ifTrue: [mejorCriptomoneda]
ifFalse: [criptomonedaActual]
].
```

3. Diagramas de clase

Para representar mi solución, la dividí en cuatro diagramas, donde se pueden observar las relaciones entre las clases. Aunque el trabajo fue implementado en Smalltalk, los diagramas son genéricos y podrían aplicarse a otros lenguajes de programación. Algunos nombres de métodos fueron abreviados para mejorar la legibilidad del diagrama.

4. Diagramas de secuencia

Para los diagramas de secuencia voy a mostrar la interacción entre las clases que se ejecutan para este Test

```
test04EntreUnRecomendadorYOtroDetectaAlQuePrediceUnMayorValor
| divulgadorMayorGananciaEsperado divulgadorMayorGananciaObtenido algoPrediccionCripto |

"Arrange"
divulgadorMayorGananciaEsperado := 'Matias'.
algoPrediccionCripto := AlgoPrediccionCripto conCriterio: 'Maximo'.

algoPrediccionCripto registrarTokenConNombre: '$PLEBEYO' ConValorActual: 10 ConRecomendador: 'M
yPatrimonio: 50000 deNacionalidad: 'Argentina' yColorDePelo: 'Colorado'.
```

```
algoPrediccionCripto registrarTokenConNombre: '$TESLA' ConValorActual: 1000 ConRecomendador: 'E'
yPatrimonio: 320000000 deNacionalidad: 'Sudafrica' yColorDePelo: 'Morocho'.
```

```
"Act"
```

```
divulgadorMayorGananciaObtenido := algoPrediccionCripto obtenerDivulgadorSegunCriterio.
```

```
"Assert"
```

```
self assert: divulgadorMayorGananciaEsperado equals: divulgadorMayorGananciaObtenido
```

para mayor claridad agrego algunos comentarios que no agrege al grafico para que no este muy cargado

- luego en el test se registra a otro recomendador

- en el test se llama a algoPrediccionCripto obtenerDivulgadorSegunCriterio.

En este diagrama se muestra como cada divulgador busca su mejor prediccion posible con sus criptomonedas segun un criterio

- Por ultimo en este diagrama se muestra como el divulgador hace las predicciones