



PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE
ESCUELA DE INGENIERÍA
DEPARTAMENTO DE CIENCIA DE LA COMPUTACIÓN

IIC2233 - Programación Avanzada
2° semestre 2015

Actividad 03

Polimorfismo

Instrucciones

En computación los números decimales son representados como un número de punto flotante o `float`, cuyo valor es una aproximación a este decimal. En algunas ocasiones conviene trabajar directamente con el número racional para mejorar la precisión y evitar pérdidas debido a la aproximación de punto flotante. Por ejemplo, el racional $1/3$ es representado como el decimal $0,333333 = 0,\bar{3}$ cuya precisión dependerá de la representación permitida por el `float`. En esta actividad usted deberá crear una clase que le permita trabajar directamente con los números racionales y efectuar con ellos todas las operaciones aritméticas y lógicas.

Requerimientos

- Crear una clase `Rational` que sea capaz de instanciarse a partir de dos enteros que representen una fracción. eg: `Rational(12,5)` \rightarrow $12/5$
- Implementar el o los métodos necesarios para poder imprimir este tipo de dato de una forma entendible. Es importante que esta representación muestre el número de una forma simplificada. eg:

- `print(Rational(24,10))` \rightarrow $12/5$
- `print(Rational(40,10))` \rightarrow 4
- `print(Rational(2,-10))` \rightarrow $-1/5$
- `print(Rational(-5,-2))` \rightarrow $5/2$
- `print([Rational(1,1), Rational(-2,3)])` \rightarrow `[Rational(1), Rational(-2/3)]`

- Implementar los métodos necesarios para poder sumar, restar, multiplicar y dividir objetos de este tipo. algunos ejemplos:

```
>>> r1 = Rational(26, 4)
>>> r2 = Rational(-2, 6)
>>> r3 = Rational(34, 7)
>>> print(r1 + r2)
37/6
>>> print(r1 * r3)
221/7
```

- Implementar los métodos para hacer comparaciones (menor, menor igual, igual, mayor igual, mayor).

Notas

- Junto con el enunciado encontrarán un archivo *main.py* donde podrán ver un código de lo que su clase debería ser capaz de hacer.
- Los métodos de suma, resta, multiplicación y división **no** tienen que modificar el objeto actual, sino que deben retornar un nuevo objeto con el valor correcto.

To - DO

- (1.00 pts) Correcta definición e instanciación de la clase **Rational**. Debe manejar bien la simplificación.
- (1.00 pts) Implementar correctamente los métodos para imprimir la clase **Rational**.
- (3.00 pts) Implementar los operadores de la clase **Rational** y que funcionen correctamente con otros objetos de tipo **Rational**
- (1.00 pts) Implementar correctamente los métodos para comparar objetos de tipo **Rational**.

Tips

- En varias de las funcionalidades que se le piden implementar se pueden aprovechar otros métodos que ya ha definido, por ejemplo la suma y la resta implementan una lógica muy similar.
- Puede crear otros métodos privados en la clase si lo siente necesario.
- Para que se puedan realizar las comparaciones entre objetos de tipo **Rational**, no es necesario que usted implemente los cinco métodos de comparación. Python es suficientemente inteligente como para inferir las comparaciones restantes (si es que posee suficientes métodos definidos).