



Programación 4

Guía Semana 2 (11/3)

InCo, FING, Udelar

Objetivo

El objetivo de esta semana es finalizar con la introducción de **conceptos básicos** de **Orientación a Objetos (OO)** y su implementación en **C++**.



Resumen :: Conceptos OO

La **herencia** es el mecanismo por el cual se permite compartir propiedades entre una clase y sus descendientes.

El **polimorfismo** es la capacidad de asociar diferentes métodos a la misma operación.

Una **operación abstracta** es una operación sin un método asociado. Eso genera una **clase abstracta**, de la cual no puede haber objetos directos.



Resumen :: Conceptos OO

```
class Numero {
    // operación sin método (abstracta), la clase es abstracta
    public: virtual float tofloat() = 0;
};

// Racional hereda de Numero sus propiedades
class Racional : public Numero {
    private: int numerador, denominador;
    public: float toFloat ();
};

// Polimorfismo permite asociar un método
float Racional::toFloat(){
    if(denominador == 0) return NAN;
    else return numerador / (float) denominador;
}

class Entero: public Numero {
    private: int valor;
    public: float toFloat ();
}

float Entero::toFloat(){ return (float) valor; }
```



Resumen :: Conceptos OO

Con la existencia de la herencia, el polimorfismo y las clases y operaciones abstractas, es necesario definir qué método se ejecutará al invocar una operación (**despacho**).

```
Numero *x = new Racional(1,2);
```

TipoEstatico(t) = Numero

TipoDinamico(t) = Racional

Resumidamente, el compilador busca la operación en el tipo estático (o en sus ancestros). Si la operación no es virtual, se despacha el método de la clase que lo declaró (**despacho estático**). Si la operación es virtual, se despacha el método según el tipo dinámico (**despacho dinámico**).



Resumen :: Conceptos OO

```
Numero *arr[3];  
arr[0] = new Racional(1,2);  
arr[1] = new Entero(2);  
arr[2] = new Racional(1,4);
```

```
float total = 0.0;  
for(int i = 0; i < 3; i++){  
    total += (arr[i])->toFloat();  
}
```

Despacha dinámicamente al método correcto, devolviendo el valor correcto

```
arr[0]->toFloat(); // Despacha el método de Racional  
arr[1]->toFloat(); // Despacha el método de Entero  
arr[2]->toFloat(); // Despacha el método de Racional
```



¿Qué hago esta semana?

1. Estudio los materiales de [Teórico](#) y las lecturas recomendadas. Las clases correspondientes se encuentran en [OpenFing](#).
03 - Conceptos Básicos de OO (desde diap.56)
2. Comienzo con el [Práctico 1](#) “Conceptos básicos de OO y C++”. Están publicadas las notas de resolución de los Ejercicios 4 y 5.
3. Preparo el ambiente de [tecnologías](#):
IDE/GitLab/modelador
4. Comienzo a trabajar en el Laboratorio 0 “Conceptos Básicos” (Disponible el 13 de marzo)

