

# Orientación a Objetos I

Explicación de práctica  
Semana del 11 de septiembre



FACULTAD DE INFORMATICA



UNIVERSIDAD  
NACIONAL  
DE LA PLATA

# Temario

- Actividades de la semana anterior
- Actividades a realizar esta semana



FACULTAD DE INFORMATICA



UNIVERSIDAD  
NACIONAL  
DE LA PLATA

# Actividades de la semana anterior

- Ejercicio 3 (Presupuestos) y haberlo testado con los tests provistos
- Ejercicio 3 Bis (Balanza Mejorada)
- Ejercicio 4 (Figuras y cuerpos) y haberlo testado con los tests provistos

## Cuadernillo Semestral de Actividades

Actualizado: 9 de septiembre de 2023

El presente cuadernillo posee un compilado con todos los ejercicios que se usarán durante el semestre en la asignatura. Los ejercicios están organizados en forma secuencial, siguiendo los contenidos que se van viendo en la materia.

Cada semana les indicaremos cuales son los ejercicios en los que deberían enfocarse para estar al día y algunos de ellos serán discutidos en la explicación de práctica.

### Recomendación importante:

Los contenidos de la materia se incorporan y fijan mejor cuando uno intenta aplicarlos - **no alcanza con ver un ejercicio resuelto por alguien más**. Para sacar el máximo provecho de los ejercicios, es importante que asistan a las consultas de práctica habiendo intentado resolverlos (tanto como les sea posible). De esa manera podrán hacer consultas más



FACULTAD DE INFORMATICA



UNIVERSIDAD  
NACIONAL  
DE LA PLATA

# Actividades esperadas para esta semana

- Ejercicio 5: Genealogía Salvaje
- Ejercicio 6: Red de Alumbrado
- Ejercicio 7: Method lookup con Empleados



FACULTAD DE INFORMATICA



UNIVERSIDAD  
NACIONAL  
DE LA PLATA

# Ejercicio 5: Genealogía Salvaje

- **Ejercicio 5: Genealogía Salvaje**
- Ejercicio 6: Red de Alumbrado
- Ejercicio 7: Method lookup con Empleados

## Ejercicio 5: Genealogía salvaje

En una reserva de vida salvaje (como la estación de cría ECAS, en el camino Centenario), los cuidadores quieren llevar registro detallado de los animales que cuidan y sus familias. Para ello nos han pedido ayuda. Debemos:

a) Modelar en objetos y programar la clase Mamífero (como subclase de Object). El siguiente diagrama de clases (incompleto) nos da una idea de los mensajes que un mamífero entiende. *Deje tieneComoAncestroA para el final y discuta su solución con el ayudante.*



FACULTAD DE INFORMATICA



UNIVERSIDAD  
NACIONAL  
DE LA PLATA

# Ejercicio 5: Genealogía Salvaje

- **Ejercicio 5: Genealogía Salvaje**
- Ejercicio 6: Red de Alumbrado
- Ejercicio 7: Method lookup con Empleados



FACULTAD DE INFORMATICA



UNIVERSIDAD  
NACIONAL  
DE LA PLATA

# Ejercicio 5 - Genealogía Salvaje

C Mamifero
-?:?
<pre>+getIdentificador(): String +setIndentificador(id: String) +getEspecie(): String +setEspecie(especie: String) +getFechaNacimiento(): Date +setFechaNacimiento(fecha: Date) +getPadre(): Mamifero +setPadre(padre: Mamifero) +getMadre(): Mamifero +setMadre(madre: Mamifero) +getAbueloMaterno(): Mamifero +getAbuelaMaterna(): Mamifero +getAbueloPaterno(): Mamifero +getAbuelaPaterna(): Mamifero +tieneComoAncestroA(unMamifero: Mamifero): Boolean</pre>

## Clase Mamifero

¿Atributos?

¿Relaciones?

¿Variables de instancia?



FACULTAD DE INFORMATICA



UNIVERSIDAD  
NACIONAL  
DE LA PLATA

# Ejercicio 5 - Genealogía Salvaje

```
public Boolean tieneComoAncestroA(Mamifero otroMamifero){  
  
    return ... ;  
  
}
```



FACULTAD DE INFORMATICA



UNIVERSIDAD  
NACIONAL  
DE LA PLATA



# Ejercicio 6: Red de Alumbrado

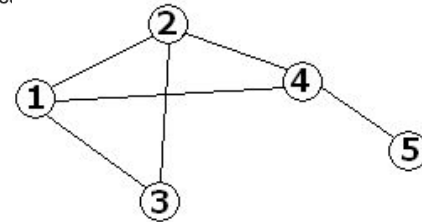
- Ejercicio 5: Genealogía Salvaje
- **Ejercicio 6: Red de Alumbrado**
- Ejercicio 7: Method lookup con Empleados

## Ejercicio 6: Red de Alumbrado

Imagine una red de alumbrado donde cada farola está conectada a una o varias vecinas formando un [grafo conexo](#)<sup>2</sup>. Cada una de las farolas tiene un interruptor. Es suficiente con encender o apagar una farola cualquiera para que se enciendan o apaguen todas las demás. Sin embargo, si se intenta apagar una farola apagada (o si se intenta encender una farola encendida) no habrá ningún efecto, ya que no se propagará esta acción hacia las vecinas.

La funcionalidad a proveer permite:

1. crear farolas (inicialmente están apagadas)
2. conectar farolas a tantas vecinas como uno quiera (las conexiones son bi-direccionales)
3. encender una farola (y obtener el efecto antes descrito)
4. apagar una farola (y obtener el efecto antes descrito)



FACULTAD DE INFORMÁTICA



UNIVERSIDAD  
NACIONAL  
DE LA PLATA

# Ejercicio 6: Red de Alumbrado

```
public class Farola{
```

```
    // Variables de instancia??
```

```
    /*
```

Crea la relación de vecinos entre las farolas. La relación de vecinos entre las farolas es recíproca, es decir el receptor del mensaje será vecino de otraFarola, al igual que otraFarola también se convertirá en vecina del receptor del mensaje

```
    */
```

```
    public void pairWithNeighbor( Farola otraFarola ){
```

```
        ...
```

```
    }
```

```
    /*
```

\* Si la farola no está encendida, la enciende y propaga la acción.

```
    */
```

```
    public void turnOn(){
```

```
        ...
```

```
    }
```



FACULTAD DE INFORMATICA



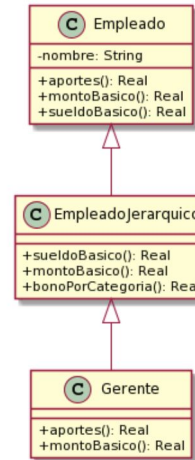
UNIVERSIDAD  
NACIONAL  
DE LA PLATA

# Ejercicio 7: Method lookup con Empleados

- Ejercicio 5: Genealogía Salvaje
- Ejercicio 6: Red de Alumbrado
- **Ejercicio 7: Method lookup con Empleados**

## Ejercicio 7: Method lookup con Empleados

Sea la jerarquía de `Empleado` como muestra la figura de la izquierda, cuya implementación de referencia se incluye en la tabla de la derecha.



Empleado	EmpleadoJerarquico	Gerente
<pre>public double montoBasico() {     return 35000; }</pre>	<pre>public double sueldoBasico() {     return     super.sueldoBasico()+     this.bonoPorCategoria(); }</pre>	<pre>public double aportes() {     return     this.montoBasico() *     0.05d; }</pre>
<pre>public double aportes(){     return 13500; }</pre>	<pre>public double montoBasico() {     return 45000; }</pre>	<pre>public double montoBasico() {     return 57000; }</pre>
<pre>public double sueldoBasico() {     return     this.montoBasico() +     this.aportes(); }</pre>	<pre>public double bonoPorCategoria() {     return 8000; }</pre>	



FACULTAD DE INFORMATICA



UNIVERSIDAD  
NACIONAL  
DE LA PLATA

# Foros de consulta

## Cómo preguntar en el foro

Breve guía para poder sacar el mejor provecho al foro y a la convivencia a través de las preguntas y respuestas.

### Cómo preguntar en el foro

Antes de Preguntar: Busca una respuesta por tus propios medios

Elegí el foro específico

Elegí un título apropiado para la pregunta

No envíes una solución para que la corrijan

Describir qué estás intentando hacer

Describir el problema y lo que has intentado para resolverlo

Escribir claro

No solicites respuestas a tu correo

Si no entendés la respuesta

Terminá con una breve nota de conclusión.

Evitá el "Me sumo al pedido"

## Foros



Ejercicio 3 - Presupuestos



## Foros



Ejercicio 5 - Genealogía Salvaje



Ejercicio 6: Red de Alumbrado



Ejercicio 7: Method lookup con Empleados



FACULTAD DE INFORMATICA



UNIVERSIDAD  
NACIONAL  
DE LA PLATA