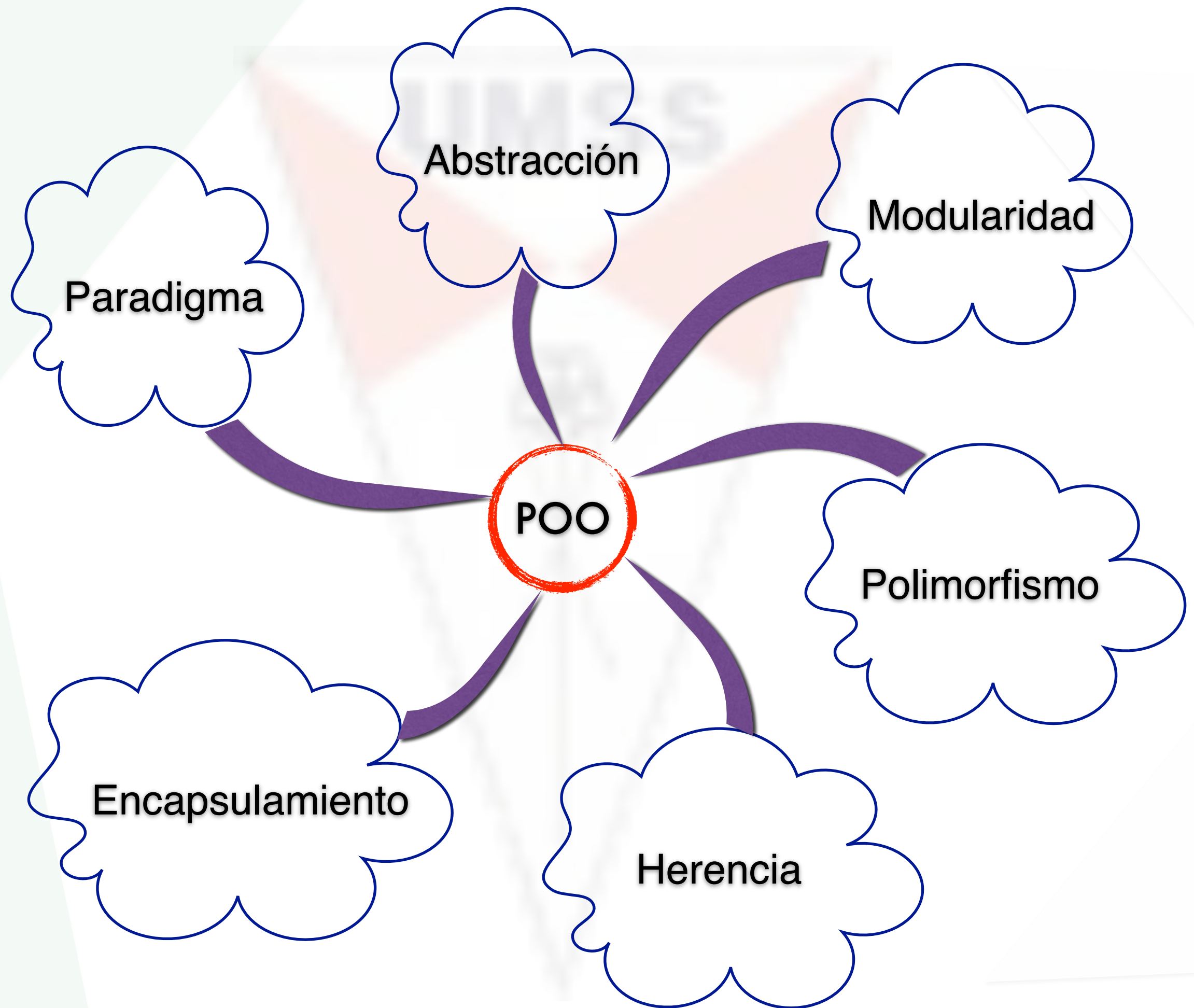
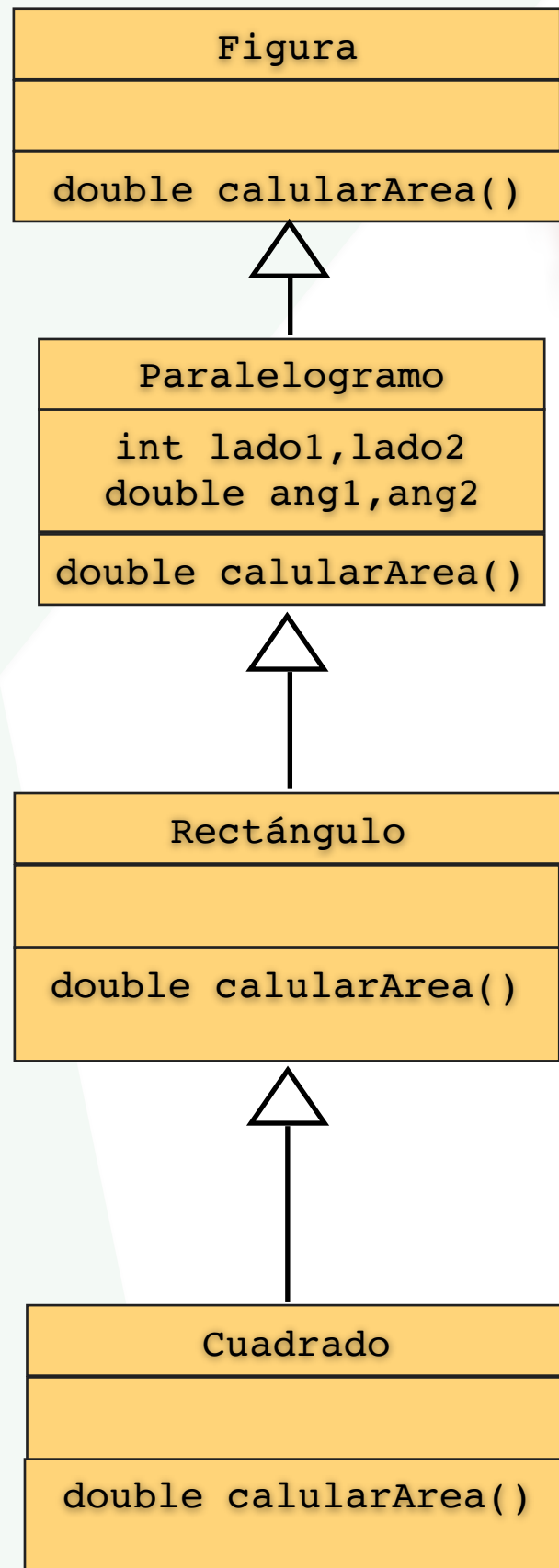




Programación Orientada a Objetos

Maria Leticia Blanco Coca





```

abstract class Figura{
    abstract public double calcularArea();
}
  
```

```

class Paralelogramo extends Figura{
    protected int lado1, lado2;
    protected double ang1, ang2;
    public Paralelogramo(int lado1, int lado2,
                        double ang1, ang2){
        this.lado1 = lado1;
        this.lado2 = lado2;
        this.ang1 = ang1;
        this.ang2 = ang2;
    }
    public double calcularArea() {
        return .....;
    }
}
  
```

```

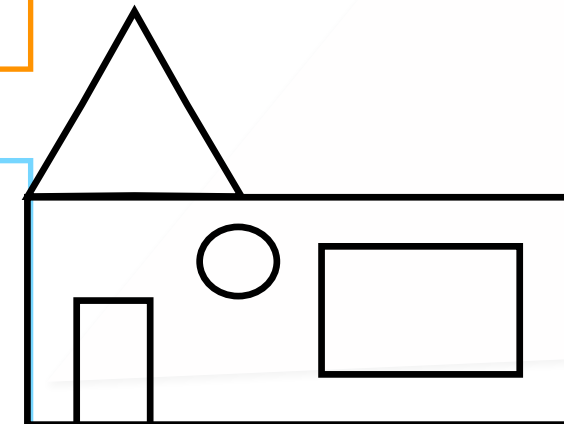
Figura obj;
obj=new Cuadrado(3);
obj.calcularArea();
obj=new Paralelogramo(3, 5, 60, 30);
obj.calcularArea();
  
```

```

class Rectangulo extends Paralelogramo{
    public Rectangulo(int lado1, int lado2){
        super(lado1, lado2, 90, 90);
    }
    public double calcularArea() {
        return lado1*lado2;
    }
}
  
```

```

class Cuadrado extends Rectangulo{
    public Cuadrado(int lado){
        super(lado, lado);
    }
}
  
```



Polimorfismo

Propiedad por la que un objeto puede asumir multiples formas y por ende comportarse de multiples formas respondiendo a un mismo mensaje. Sobreescritura, permite a las subclases redefinir la implementación de un método heredado.

- Imágenes de Grady Booch, “Object-Oriented Analysis and Design with Applications”