

Lenguajes, Tecnologías y Paradigmas de la programación (LTP)

Práctica 6: Módulos y Polimorfismo en Haskell (Parte 2: Polimorfismo ad hoc o sobrecarga)



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA

Sergio Pérez
serperu@dsic.upv.es

SESIÓN 2: Polimorfismo ad hoc o sobrecarga

OBJETIVOS DE LA PRÁCTICA

Polimorfismo ad hoc o sobrecarga:

Ver como crear nuestras propias jerarquías de tipos (Herencia):

- Utilizando la **creación de tipos de datos**
- **Creando** nuestras propias **clases de tipos nuevas**

Polimorfismo por creación de tipos de datos

```
type Side = Float  
type Apothem = Float  
type Radius = Float
```

Polimorfismo por creación de tipos de datos

```
type Side = Float  
type Apothem = Float  
type Radius = Float  
  
data Shape =
```

Polimorfismo por creación de tipos de datos

```
type Side = Float
```

```
type Apothem = Float
```

```
type Radius = Float
```

```
data Shape = Pentagon Side Apothem
```

Polimorfismo por creación de tipos de datos

```
type Side = Float
```

```
type Apothem = Float
```

```
type Radius = Float
```

```
data Shape = Pentagon Side Apothem | Circle Radius
```

Polimorfismo por creación de tipos de datos

```
type Side = Float
```

```
type Apothem = Float
```

```
type Radius = Float
```

```
data Shape = Pentagon Side Apothem | Circle Radius  
           deriving (Show,Eq)
```

Polimorfismo por creación de tipos de datos

```
type Side = Float
```

```
type Apothem = Float
```

```
type Radius = Float
```

```
data Shape = Pentagon Side Apothem | Circle Radius  
           deriving (Show,Eq)
```

```
perimeter :: Shape -> Float
```


Polimorfismo por creación de tipos de datos

```
type Side = Float
```

```
type Apothem = Float
```

```
type Radius = Float
```

```
data Shape = Pentagon Side Apothem | Circle Radius  
           deriving (Show,Eq)
```

```
perimeter :: Shape -> Float
```

```
perimeter (Pentagon s a) = 5 * s
```

Polimorfismo por creación de tipos de datos

```
type Side = Float
```

```
type Apothem = Float
```

```
type Radius = Float
```

```
data Shape = Pentagon Side Apothem | Circle Radius  
           deriving (Show,Eq)
```

```
perimeter :: Shape -> Float
```

```
perimeter (Pentagon s a) = 5 * s
```

```
perimeter (Circle r) = 2 * pi * r
```

Polimorfismo por creación de tipos de datos

- ¿Y si quiero añadir `Rectangle` `Base` `Height`?

```
type Base = Float
```

```
type Height = Float
```

Polimorfismo por creación de tipos de datos

- ¿Y si quiero añadir `Rectangle` `Base` `Height`?

```
type Base = Float
```

```
type Height = Float
```

```
data Shape = ...
```

Polimorfismo por creación de tipos de datos

- ¿Y si quiero añadir `Rectangle Base Height`?

```
type Base = Float
```

```
type Height = Float
```

```
data Shape = ... | Rectangle Base Height
```

Polimorfismo por creación de tipos de datos

- ¿Y si quiero añadir `Rectangle Base Height`?

```
type Base = Float
```

```
type Height = Float
```

```
data Shape = ... | Rectangle Base Height
```

```
perimeter :: Shape -> Float
```

```
...
```

Polimorfismo por creación de tipos de datos

- ¿Y si quiero añadir `Rectangle Base Height`?

```
type Base = Float
```

```
type Height = Float
```

```
data Shape = ... | Rectangle Base Height
```

```
perimeter :: Shape -> Float
```

```
...
```

```
perimeter (Rectangle b h) = 2 * (b + h)
```

Polimorfismo por creación de tipos de datos

- ¿Y si quiero añadir `Rectangle Base Height`?

```
type Base = Float
```

```
type Height = Float
```

```
data Shape = ... | Rectangle Base Height
```

```
perimeter :: Shape -> Float
```

```
...
```

```
perimeter (Rectangle b h) = 2 * (b + h)
```

Ampliar `Shape` implica modificar la definición actual de su tipo

Polimorfismo por creación de clases de tipos

```
type Side = Float  
type Apothem = Float  
type Radius = Float
```

Polimorfismo por creación de clases de tipos

```
type Side = Float
```

```
type Apothem = Float
```

```
type Radius = Float
```

```
data Pentagon = Pentagon Side Apothem
```

```
data Circle = Circle Radius
```

Polimorfismo por creación de clases de tipos

```
type Side = Float
```

```
type Apothem = Float
```

```
type Radius = Float
```

```
data Pentagon = Pentagon Side Apothem
```

```
data Circle = Circle Radius
```

```
class Shape a where
```

Polimorfismo por creación de clases de tipos

```
type Side = Float
type Apothem = Float
type Radius = Float

data Pentagon = Pentagon Side Apothem
data Circle = Circle Radius

class Shape a where
    perimeter :: a -> Float
```

Polimorfismo por creación de clases de tipos

```
type Side = Float
type Apothem = Float
type Radius = Float

data Pentagon = Pentagon Side Apothem
data Circle = Circle Radius

class Shape a where
    perimeter :: a -> Float
    area :: a -> Float
```

Polimorfismo por creación de clases de tipos

```
type Side = Float
```

```
type Apothem = Float
```

```
type Radius = Float
```

```
data Pentagon = Pentagon Side Apothem
```

```
data Circle = Circle Radius
```

```
class Shape a where
```

```
    perimeter :: a -> Float    → Define un comportamiento
```

```
    area :: a -> Float
```

Polimorfismo por creación de clases de tipos

```
type Side = Float  
type Apothem = Float  
type Radius = Float
```

```
data Pentagon = Pentagon Side Apothem  
data Circle = Circle Radius
```

```
class Shape a where  
  perimeter :: a -> Float  
  area :: a -> Float
```

→ Define un comportamiento
interfaces java

Polimorfismo por creación de clases de tipos

```
class Shape a where  
  perimeter :: a -> Float  
  area      :: a -> Float
```


Polimorfismo por creación de clases de tipos

```
class Shape a where  
  perimeter :: a -> Float  
  area      :: a -> Float
```

¿Cómo hago que `Pentagon` y `Circle` instancien la clase de tipos `Shape`?

Polimorfismo por creación de clases de tipos

```
class Shape a where  
  perimeter :: a -> Float  
  area :: a -> Float
```

¿Cómo hago que `Pentagon` y `Circle` instancien la clase de tipos `Shape`?

```
instance Shape Pentagon where
```

Polimorfismo por creación de clases de tipos

```
class Shape a where  
  perimeter :: a -> Float  
  area :: a -> Float
```

¿Cómo hago que `Pentagon` y `Circle` instancien la clase de tipos `Shape`?

```
instance Shape Pentagon where  
  perimeter (Pentagon s a) = 5 * s  
  area ...
```

Ejercicios

- Ejercicios Parte 2:
 - Ejercicios 6 - 9