

# Ejercicios

Consideremos el tipo `Nat` que hemos definido anteriormente. Se pide:

1. sobrecargar los operadores aritméticos (`+` y `*`) de la clase `Num` para poder usarlos para sumar y multiplicar naturales

2. sobrecargar `show` de la clase `Show` para mostrar los naturales como el número que representan, por ejemplo `Zero` como `0`, `Suc Zero` como `1`, ...

3. sobrecargar los métodos necesarios de `Enum` para poder definir listas aritméticas con valores de tipo `Nat`. Por ejemplo: `[Zero..Suc (Suc Zero)]` daría `[Zero, Suc Zero, Suc (Suc Zero)]`

4. Sobrecargar `<` de la clase `Ord` para poder comparar valores del tipo `Nat` según el valor de su área

5. Sobrecargar `show` de la clase `Show` para visualizar un círculo con su radio entre paréntesis (e.g., `(2.5)`) y un rectángulo con sus lados entre corchetes separados por una coma (e.g., `[1.5,2.5]`).