

Atividade 7

Paradigmas de Programação

Arthur Pellenz Winck

00674545001

1) As classes primitivas são aquelas que não derivam de outras classes, ou seja, não são formadas a partir de outras classes. Exemplos de classes primitivas são:

- Eq: realiza todo o tratamento dos métodos de igualdade ou desigualdade, instanciando todos os tipos, exceto aqueles de input/output e exceções. Os seus métodos são “==” ou “/=". Eles basicamente retornam o inverso do outro método.
- Enum realiza o tratamento da enumerabilidade, definindo os métodos de operações sobre tipos sequencialmente ordenados como (), Bool, Char, Int e Integer, entre outros. Alguns dos seus métodos são sucessor e predecessor, a função de converter um inteiro em um Enum e o processo inverso, de retornar um inteiro recebendo um Enum.
- Show é uma das classes que define os métodos de conversão entre números e caracteres e vice-versa. Ela instancia todos os tipos exceto IO e funções. Ela possui alguns métodos como show, nele ela recebe um item e retorna uma string dele, e o showList que recebe uma lista de elementos e retorna a lista visualizável desses elementos em string.

2) Classes secundárias são as classes que foram definidas a partir de outras classes.

- Num é uma classe que deriva de Eq e Show, e nela temos alguns métodos como: somar, subtrair, multiplicar, negar e realizar o módulo do número, transformar um inteiro em um Num. Por exemplo, o método de subtração é realizado negando o número que será negativo na operação. Para negar um número, é realizada a subtração desse número de zero. Num instancia os tipos Int, Integer, Float e Double
- Real é uma classe que deriva de Num e Ord, e instancia Int, Integer, Float e Double. Ele possui um método que transforma um elemento em um número racional, do tipo Rational

3) Além das classes Num e Real, também existem outros tipos numéricos, como:

- Integral, ele deriva de Real e Enum e possui diversos métodos: quot e rem (quociente e sobra), divisão e mod. Além disso possui métodos para retornar uma lista de elementos como quotRem (quociente e sobra) e divMod (divisão e mod). Além disso, possui o método de converter para um Inteiro, onde recebe um elemento e retorna um Integer.

- Fractional é uma subclasse de Num que possui o método de divisão (/), o método recip que retorna o inverso (1/argumento) e o método para transformar um item de Rational para Fractional.
 - Floating deriva de Fractional e possui alguns métodos como: pi (retorna o número de pi), exp (expoente), log e sqrt (o quadrado). Também existe a operação de potenciação (**) e logBase que adiciona a base para realizar o cálculo do log. Além disso, também existem as operações trigonométricas.
 - RealFrac é uma classe que deriva de Real e Fractional e possui alguns métodos como properFraction que retorna o número inteiro e a sua fração (que é menor que 1) e os métodos de arredondamento: truncate, round, ceiling e floor.
 - RealFloat é uma classe que deriva de RealFrac e Floating e tem muitos métodos como: floatRadix, floatDigits e floatRange.
- 4) Polimorfismo é a capacidade de um objeto em decidir que método aplicar a si mesmo. Polimorfismos AdHoc tem um número finito de variações e tem como tipos o Polimorfismo com Overloading e polimorfismo com coerção. Ele é resolvido estaticamente em tempo de compilação. O polimorfismo com overloading, ou sobrecarga, é basicamente mudar qual método será executado pela quantidade/tipo dos parâmetros. Polimorfismo de coerção é quando um tipo é convertido para outro implícita ou explicitamente.

Polimorfismo universal possui dois sub-tipos: parametrização e inclusão. O polimorfismo de parametrização é basicamente a criação de um método que pode ser escrito genericamente e especificado quando é criado. O polimorfismo de inclusão utiliza a capacidade de substituição da herança para permitir comportamentos diferentes dependendo do tipo (classe pai ou classe filha) de quem chama o método.