Functional Programming with Scala: Class 8

John Nestor
47 Degrees, Persist Software

uw@persist.com

December 5, 2016

Review

- Futures
  - Present solution(s)
  - Code review and discussion

Lecture

DSLs

- External (use Scala to implement)
  - Parsing combinators - slow
  - Use Java parser instead: ANTLR (LL), Beaver(LALR)
- Internal (add as part of Scala)
  - Operator definitions
  - Macros
  - Implicits
  - Type system features

Hazards of DSLs

- Hard to create
- Strange syntax
- Confusing implicits
- Poor error messages
- Maintainability
- Abandonment

Features Used in DSLs

- Operator names, prefix and infix operators
- Implicits: class, parameter, conversion
- Default parameter values and by name parameters
- Apply/unapply
- Macros

Macros

- Documentation (experimental!)
- Must be compiled separate from use
- Definition function references implementation
- Implementation
  - must be at top level
  - uses internal compiler parse tree
  - supports type parameters via type tags
- Quasiquotes
  - Easy way to build parse tree in macros
  - Can be used to insert computed values
  - Variable length list support via ..

Kinds of Macros

- Macro Examples
- Blackbox
    - function with fixed return type
    - look at sample code
- Whitebox
    - function with dynamic return type
    - type providers (function returns instance of new type)
    - not handled well in Intellij (use eclipse)
    - look at sample code

DSL’s

- SBT
- Spark
- Shapeless (uses Whitebox macros)
- Cats (functional programming)
- Cypher (this week coding assignment)

LambdaTest

- lambdatest package
    - implicit conversion
    - implicit parameter
    - by name param
    - default param value
    - multiple parameter lists
    - type parameters
- LambdaAct
    - infix operator

Units

- Physical unit checking
- U
  - infix and postfix operators
  - type parameters
  - type bounds
  - implicit parameters
- Integers
  - type declarations with type parameters

Assignment 8

Assign 8: Cypher DSL

- Goals
  - Implement a DSL
  - Look at another graph algorithm
  - See examples of operator definition
  - See and use for comprehensions

Graph of the Gods

- Titan example
- http://s3.thinkaurelius.com/docs/titan/1.0.0/getting-started.html
- Look at code
    - object Graph in Cypher.scala
    - test file GodGraph.scala
    - look at ??? in class Graph in Cypher.scala
    - look at first example in test file CyperTest.scala

Cypher

- Query language for Neo4j
  - http://neo4j.com/developer/cypher-query-language/
  - look at other examples in test file CyperTest.scala
  - look at ??? code in object Cypher in Cypher.scala

Assignment

- Implement all the ??? places
- Make sure all tests pass
- Hint: missing cypher implementations can be implemented as for comprehensions