

函数式编程原理

课程报告

**姓 名：冯就康**

**学 号：I201920029**

**班 级：CS2002**

**指导教师：顾琳**

**计算机科学与技术学院**

**2023 年 10 月 16日**

一、函数式语言家族成员调研

完成函数式语言家族成员调研报告，内容可包括但不限于：庞大的函数式语言家族中有哪些成员？都由谁提出来的？各自有什么特征？没落和兴盛的原因？

**函数式编程语言家族调研报告**

**引言**

函数式编程语言，以数学函数为基础，强调程序执行的结果而非执行的过程，具有强大的表达能力和优雅的抽象模型。这篇报告将对函数式编程语言家族进行详细的调研。

**函数式编程语言家族成员及其发起者**

1. **LISP**: 由John McCarthy在1958年提出，它是最早的函数式编程语言，也是AI研究的基石。
2. **Haskell**: 由Simon Peyton Jones等人在1990年提出，是纯函数式编程语言，被广泛用于教学和研究。
3. **Erlang**: 由Joe Armstrong在1986年提出，主要应用于并行、分布式系统和实时系统。
4. **Scala**: 由Martin Odersky在2004年提出，结合了面向对象和函数式编程特性，主要应用于并行计算和大数据处理。
5. **Clojure**: 由Rich Hickey在2007年提出，是一种运行在Java平台的现代、实用的LISP方言。
6. **Elixir**: 由José Valim在2011年提出，运行在Erlang VM上，主要用于构建分布式、容错性强的系统。

**各成员特征**

1. **LISP**: 以列表作为主要数据结构，以S表达式作为基本语法。
2. **Haskell**: 支持类型推导，惰性评估和纯函数式，有强大的类型系统。
3. **Erlang**: 设计目标是构建容错性强、低延迟的大规模并行系统。
4. **Scala**: 有丰富的类型系统，同时支持面向对象和函数式编程。
5. **Clojure**: 强调不可变性和函数式编程，有丰富的数据结构和宏系统。
6. **Elixir**: 简洁优雅的语法，良好的并行和分布式支持。

**兴盛和没落的原因**

函数式编程语言的兴盛主要受益于对并行计算、大数据处理和响应式编程的需求增长。例如，Scala和Elixir在这些领域都表现出了显著的优势。

至于没落，一部分原因是函数式编程语言的学习曲线较陡峭，对许多开发者来说，理解和掌握函数式编程需要花费大量的时间和精力。此外，市场对于函数式编程的需求相比命令式编程还是有所不足。

**结论**

函数式编程语言家族的成员多种多样，每种语言都有其独特的特性和应用领域。虽然函数式编程语言在市场需求和学习曲线上面临一些挑战，但其对于并行计算、大数据和响应式编程的优势使其在某些领域拥有无可比拟的优势。在未来，我们有理由相信，随着技术的发展和市场需求的变化，函数式编程将会发挥出更大的作用。

二、上机实验心得体会

在熟悉安装SML/NJ开发环境的基础上，完成实验内容。在完成实验中选取2-3个感受最深的实验，谈谈在函数式编程学习和实践中的心得和体会。**内容包括但不限于结题**思路、代码、运行结果、性能分析、遇到的问题及如何解决等。

三、课程建议和意见

课程知识梳理总结或课程建议，包括头歌平台实验部署的改进意见和方案。