# 函数式编程报告

* 实验总结

实验一很简单，主要考查了对SML使用方法的了解，知道ml程序在编译器中是如何编译运行的，掌握sml基本语法和简单程序设计。有效的算法可以帮我减少一半的代码。实验二需要掌握ml语言的表list结构和树tree结构，学会分析设计程序的性能，我也学会如何构造并遍历树结点，如何使用系统的比较函数。实验三主要牵扯到高级函数，函数式编程语言有很多特点诸如引用透明性，高级函数，惰性求值，递归调用，类型检测等，它的函数可以当做数据，参数，返回值来使用。高级函数主要包括合成和柯里化（即函数只能有一个参数，对于多个参数的函数，把它们当做元组输入，也可以实现为一个函数返回另一个函数作为结果，称为柯里化），从而提高了函数实用性和适用范围。

* 课后感想

不知不觉就被老师带着把函数式程序设计和ML语言了解了一番，实验时一番操作猛如虎，结果还是被实验虐了一遍，不是被它的难度，更多的是被它的设计和实现方式，方能体会到，原来同样的问题还可以这样解决。

ML语言是函数式语言，这种基于数学概念，背离了我们过去所学习和习惯的设计理念和方式的设计语言，它更多地像是一堆富有逻辑性的函数（对我而言）聚集起来，当你遇到问题，就设计几个函数，或作为参数，或作为返回值，或者就是函数，就可以以强逻辑性而又容易理解的解决你的问题。ML语言综合了数学概念和机器语言，根据作者的说法，是为了尽可能的把我们从过程繁琐的程序设计中走出来，同时避开高深数学模型的理解执行。

比起高级语言的各种各样的库、函数，ML语言的设计更加简单，它大量的使用各种函数，减少了代码重复，进而达到提高编程效率的目的。我喜欢ML语言的‘：：’和‘@’，简单易懂，又方便实用，作者热衷于使用多种符号和口语化的语言来设计ML。除此之外，ML语言设计的多态类型检测，抛出和捕获异常等功能都为它被称为证明策略编程的元语言提供了强有力的帮助。

总而言之，学习ML语言改变了我对程序设计的认知，原来程序还可以理解，这样写！