

Week2 Paper Summary——Recursive Functions of Symbolic Expressions

By Yi Zhou

This week's paper by John McCarthy is also considered as the foundation of Programming Language, especially for functional programming language, such as Lisp.

It comes up with two novel concepts——S-expressions and S-functions, which is based on the lambda-calculus raised by Church. The paper first shows us the mathematical concept of functions and then uses S-expressions and S-functions to symbolize it. Just by combining some basic notation, such as axioms, such a system can come out with quite complicated functions, especially recursive ones by using combination of S-expressions and S-functions. And then it shows us how to realize these concepts in LISP system.

As far as I am concerned, I'm quite surprised with the elegance of S-expressions and S-functions. It's quite amazing that recursive functions and some other complex ones can be explained clearly by just combining such simple things. Functional programming is quite an elegant style of programming based on mathematics. However, it could be quite confused for beginners to get such ideas. Although, McCarthy tried to make the expression more readable, it could still be complex for novices and that may be due to the complexity of mathematics itself. But, admittedly, there is no denying that functional programming provides us with a brand new method to describe the world.