ラムダ計算入門

Sakura Math

2021 年 9 月 13 日

1　概要

ラムダ計算は函数概念を用いた計算が持っている本質的な性質を抽象した概念です．本講演ではラムダ計算の初歩から始め，どの程度の計算能力を有するかということを緩く紹介したいと考えています．入門枠ということもありますので，形式的な議論を極力減らし，具体例を通して実際に計算できることを納得してもらえればと考えています……．困りましたね，これ以上書くことがありません．仕方がないので種を明かしてしまおうと思います．

2　種

まずは自然言語の水準で素朴に函数概念を定義しようとすると，例えば次のようになるでしょう：値を指定すると，指定された値に基づいて一定の計算結果を返すような函である．これは function に函数なる訳語が当てられた頃の認識を踏まえても大凡正しいもののようです．

更に一歩踏み込んで，値を指定するたびに一定の計算結果を返すということの意味を考えてみると，まず具体的な値に函数を**適用**し，それを計算するために不定元 x が書かれている場所に指定した値を**代入**することになります．その上で代入後の x を含まない式を**評価**していくことになりますが，特に**評価**する段階に於いては最早不定元は存在しません．よってその**評価**を実行するためには a priori に評価方法が定義されていなければならないことに注意しましょう．

こういったことを踏まえると，計算としての函数概念は a priori に定義されている評価方法を不定元を用いて**抽象**することで得られる函であり，その函に備わっている本質的な機能は，具体的な値に函数を**適用**することと適用した函数へ値を**代入**することのみに過ぎないといった考えに至ります．ここまでで述べた**適用**と**代入**なる二つの機能のみを記号列によって実現するのがラムダ計算になりますが，一見すると a priori に評価方法が定義されていなければただの木偶に過ぎないように思われます．併し決してそんなことはなく，一定の記号列を**符号化**して自然数だと見做すことによって，ラムダ計算の範疇に於いて自然数の加法や乗法などの**評価**を実現できることが判明します．

3　補足

本講演で扱うラムダ計算は純粋なものに限り，型がないものを扱います．目標はラムダ計算の定義，不動点コンビネータ，Church 数，計算可能な函数が実行できそうであるということを理解することを目標にします．