

# 情報数学 最終課題

72043913/t20391ks 澤田 開杜

## 概要

情報数学の授業で学んだラムダ計算について下記にまとめて記す。

## テーマを選んだ動機

最近のプログラミング言語では、HaskellやScalaのような関数型言語でなくとも、関数が第一級オブジェクトとなっているような言語が多い。私もこの授業を受ける前までは、λ式というと"無名関数"・"関数オブジェクト"といったような認識でしかいなかった。しかし、学んでみると普通のプログラム言語のような表現力を持っており、奥が深く、とても興味を持ったためである。

## ラムダ計算

### ラムダ記法

ラムダ式とは、いわば名前の無い関数である。例えば+の演算子がすでに定義されているとして、引数を1つ取り、それに1を加える関数を定義すると、 $add1(x) = x + 1$ のように記述するだろう。しかし、関数の名前自体は本質的ではない。そこで、この関数を $\lambda x. x + 1$ と表現することにする。"λ"という記号の直後に引数を記述し、"."の後に関数の本体を記述する。このように仮引数を指定し、式から直接関数を定義する操作をλ抽象という。このように定義したλ式を $add1(3)$ のように関数を値に適用するには、 $(\lambda x. x + 1)(3)$ のように記述する。

### カーリー化

先程は $add1$ という、引数が一つの関数をもとにラムダ式での形を考えた。ここで新しく $add(x, y) = x + y$ という2つの引数を取る関数を考え、それをλ抽象した結果を考えるが、λ式では、一般に複数引数を受け取る形では記述しない。そのため、これをλ抽象するにはカーリー化という手法が大切になる。関数 $add$ は2つの数を受け取って1つの数を返すため、型は $N \times N \rightarrow N$ である。簡単に説明すると、カーリー化とはこの $N \times N \rightarrow N$ のような関数を $N \rightarrow N \rightarrow N$ のような形にすることである。