付録 B

型付きラムダ計算の停止性

本章はラムダ計算に型を導入し，この体系の性質のうち，特に強停止性を詳しく調べる．抽象書換え系においては，書換えの元の構造は考えず，書換えの関係のみの情報から計算の性質を考察したが，本章では項の構造とそれと密接に関連した書換えの関係を帰納法を多用して詳しく考察する．このような計算過程の厳密な考察により「計算とは何か」という問いに対する新たな理解を得ることを目的とする．

B.1　ラムダ計算への型の導入

4 章で導入したプログラミング言語 $\mathcal{L}$ は自然数のみをデータとして扱う言語であった．データが自然数以外の多様なものであっても，ゲーデル符合化によりデータを自然数に符号化できるため，計算の能力を理論的に議論するにはデータを自然数に限定してもよい．しかし，現実に幅広く使われるプログラミング言語では，プログラムを理解しやすくする，プログラムの実行速度を向上させる，誤りの少ないプログラムを作るなどの理由から，多様なデータを扱うことができるようになっている．

多くのプログラミング言語はまた，扱うデータの属している集合を明確に定めるための「型」の概念を持っている．たとえば，C 言語や Java 言語には，整数の型 int や，文字型 char がある．プログラミング言語に型を導入すると，プログラムが分かりやすくなる．さらに，型を検査することにより，プログラミング作成時の間違いの予防や事前の検出ができるという利点がある．また，データの型があらかじめ決まっていれば，プログラムのより高速な実行も可能になる．このような利点は理論的な計算の能力を向上させることには直接つながらないが，計算を形式化する際には，記述されたものの明晰さや安全さは重要な事柄である．

7 章で考察したラムダ計算は λ 項を計算の対象とする体系であった．ここにおいても，プログラミング言語 $\mathcal{L}$ と同様な発想で，必要なデータを簡略にするという設定が行われていると見れる．すなわち，ラムダ計算は λ 項のみをデータとして扱う体系であったが，自然数を λ 項に符号化することができるために，λ 項のみで帰納関数との計算の能力の同等性を調べることができた．だが，プログラミング言語の場合と同様に，もしこのような符号化をせずに，直接ラムダ計算で自然数が扱えれば，計算の理論と実際の計算の解析を繋ぐ橋渡しになる．

私たちは 7 章までで，チューリング機械，帰納関数，ラムダ計算を用いて計算の能力の解析を行い，その等価性をみてきた．本章ではこのような計算の能力を限界を理解した後の次のステップとして，計算の理論の厳密さを保ちつつ人間が計算をより明確に把握する方法をラムダ計算を対象にして探る．すなわち，ラムダ計算に型の概念を取り入れた型付きラムダ計算の体系を考察する．特に 7 章のラムダ計算の理論を基礎として，6 章の抽象書き換え系の理論を使い，型付きラムダ計算の性質を調べる．

この章でラムダ計算に導入する「型」は二種類ある．それらは基底型と関数型と呼ばれる．基底型とは，関数のように入力をとるようなものではなく，データの基底をなす型である．例えば整数の型 int や

文字型 char などがある．関数型は基本的にはλ 項の抽象式の型で，例えば λ 項

$$\lambda x{:}\mathsf{int}.x$$

は型 int → int という関数型を持つ．int が基底型を表し，型 int → int は，「int 型のデータをもらい int 型のデータを返す関数」の型を表す．つまりλ 項 $\lambda x : \mathsf{int}.x$ がそのような関数を表しているとみる．この λ 項は 7 章のλ 項に，変数の型の情報を付加したものである．このようにラムダ計算に型を取り入れることにより「何のデータに対して計算をする関数か」が明確になる．この明解さのため，型付きラムダ計算は関数型プログラミング言語という種類のプログラミング言語の基礎理論としても用いられている．

B.2　ラムダ計算と停止性

以降，本章で導入する型付きラムダ計算と区別するために，7 章で議論したラムダ計算を**型のないラムダ計算**と呼ぶことにする．

前節でラムダ計算に型を導入することの利点として，計算系の明解さを指摘したが，もう一つ大事な点がある．それは，型付きラムダ計算においては，どんなλ 項もβ 簡約を繰り返し行えば必ず最後には簡約できない項，正規形に辿り着くという特徴である．これは 6 章の言葉を用いれば，型付きラムダ計算がつくる抽象書換え系が強停止性をもつ（SN である）ことである．強停止性は型付きラムダ計算に特徴的な性質で，型のないラムダ計算においては停止しない簡約を起こすλ 項が存在した．例えば次の型のないラムダ計算のλ 項を考えてみよう．

$$(\lambda f.ff)(\lambda f.ff)$$

これをできる限りβ 簡約することを試みると

$$(\lambda f.ff)(\lambda f.ff) \to (\lambda f.ff)(\lambda f.ff) \to (\lambda f.ff)(\lambda f.ff) \to \cdots \tag{B.1}$$

と永遠に同じ項のままβ 簡約を繰り返しこの計算は止らない．

型付きラムダ計算では，(ff) という項は型を持たないため，(ff) や $(\lambda f.ff)(\lambda f.ff)$ は型付きラムダ計算での項ではなくなる．停止しないλ 項は他にも様々なものがあるが，本章で導入する型の概念は，このような止まらない計算を引き起こすλ 項をすべて禁止することになる．結果として型付きラムダ計算は強停止性を持つことになるが，この事実の証明は型付きラムダ項の構造と簡約の解析を精密に行うことによって示すことができる．証明は本書でこれまで扱ってきた関係の概念と帰納法を駆使して行う．型付きラムダ項の構造と簡約は，「導出木」という帰納的な構造を用いて定義されるため，これらの構造の解析には「導出に関する帰納法」という整礎帰納法の特別な場合の証明方法を頻繁に用いることになる．本章はこれを用いた型付きラムダ計算の停止性の証明を詳しく行う．

以降の議論では強停止性のみ議論するので，強停止性のことを単に**停止性**ということにする．

B.3　型 付 き λ 項

型

これから述べる型付きラムダ計算は単純型と呼ばれる基底型と関数型から構成される型の概念を持つ．本書では単純型を単に型と呼ぶことにする．型を以下のように定義する．

まず基底型の集合 $\mathcal{B}$ を適当に定め，これを用いてすべての型を構成する．

定義 B.1 (型)

(1) 基底型 $\sigma \in \mathcal{B}$ は型 σ である．

(2) 型 σ と τ から作られる $(\sigma \to \tau)$ は型である．

基底型の集合を $\mathcal{B} = \{\mathsf{int}, \mathsf{char}, \mathsf{bool}\}$ とすると，型には例えば以下のようなものがある．

$$\mathsf{int}, \qquad (\mathsf{int} \to \mathsf{bool}), \qquad (\mathsf{int} \to (\mathsf{bool} \to \mathsf{char}))$$
$$((\mathsf{int} \to (\mathsf{int} \to \mathsf{char})) \to (\mathsf{bool} \to \mathsf{int}))$$

また記号 σ, τ, ι を型を表すものとしてよく用いる．$\to$ を右に強く結合すると約束し，型を表すのに用いる括弧を適宜省略する．例えば，$(\sigma \to (\tau \to \iota))$ を $\sigma \to \tau \to \iota$ と略記する．

前項の文法

型付き λ 項はλ 項に型の情報が付いている表現であるが，ここでつける型の情報は,「正しく」なければならない．ここでの「正しさ」とは項の構成と型の情報が私たちが意図している形に整合しているか，ということである．例えば $(\lambda x{:}\mathsf{int}.\lambda y{:}\mathsf{int}.xy)$ は型付きλ 項ではない．なぜならば，変数 x は int 型であるので，作用式 xy は型が合っていない ($x : \mathsf{int} \to \sigma$ でなければならない) からである．

したがって型付きラムダ計算でのλ 項を定義するには，型のないラムダ計算の場合の定義よりももう少し複雑な手続きを踏む必要がある．まず前項という型の整合性は無視した項を初めに定義してから，それに**型割り当て**という手続きを行い型を割り当てたものを型付きλ 項と定義する．

定義 B.2 (前項)

(1) 変数 x で作られる式 x は前項である．

(2) 変数 x と型 σ と前項 M から作られる式 $(\lambda x{:}\sigma.M)$ は前項である．

(3) 前項 M と前項 N から作られる式 (MN) は前項である．

(4) 上の ((1)) から ((3)) によって作られる式のみが前項である．

前項は 7 章のλ 項とほとんど同じであるが，抽象の場合のみ異なり,「$\lambda x{:}\sigma$」のように型の情報が付くことに注意しよう．

型割り当て

変数 x と型 σ に対して，$x : \sigma$ により，変数 x が型 σ であることを表す．このとき，変数 x が型 σ を持つとも言う．これらを集めた有限集合

$$\{x_1 : \sigma_1, \ldots, x_n : \sigma_n\}$$

を**型環境**と呼ぶ．型環境の集合を表すには Γ を用いる．また $\Gamma, x : \sigma$ は $\Gamma \cup \{x{:}\sigma\}$ の略記で，かつ $x{:}\sigma \notin \Gamma$ であると約束する．

さらに前項 M と型 σ に対して，以下の表現を**型判断**と呼ぶ．

$$\Gamma \vdash M : \sigma$$

この型判断は，λ 項 M は型環境 Γ のもとで型 σ を持つと読む．ここでλ 項 M の自由な変項とその型の情報は型環境 Γ から得られる．型判断の $\vdash$ の左に書く型環境には括弧 を省略する．

正しい型判断は，次の方法で型を前項に割り当てることによって得られる．以下を型割り当て規則と呼ぶ．

定義 B.3 (型割り当て規則)

$$[\text{変項}]\,\frac{}{\Gamma, x:\sigma \vdash x:\sigma}$$

$$[\text{抽象}]\,\frac{\Gamma, x:\sigma \vdash M:\tau}{\Gamma \vdash \lambda x{:}\sigma.M : \sigma \to \tau} \qquad [\text{作用}]\,\frac{\Gamma \vdash M:\sigma \to \tau \quad \Gamma \vdash N:\sigma}{\Gamma \vdash MN:\tau}$$

これらの分数のような表現は，上段の型判断が成り立つとき，下段の型判断が成り立つ，という規則を表している．左にかいた［と］で囲んだ言葉は規則の名前を表す．規則［変項］の場合は，上段が空であるから，常に下段が成り立つ．つまり型環境に $x:\sigma$ があれば，変項 x の型は σ であると言っている．例えば，正しい型判断として以下のようなものがある．

(1)　$x:\mathsf{int}, y:\mathsf{int}, z:\mathsf{char} \vdash y:\mathsf{int}$

(2)　$x:\mathsf{int}, f:\mathsf{int} \to \mathsf{char} \vdash fx:\mathsf{char}$

(3)　$a:\mathsf{int} \vdash \lambda y{:}\mathsf{int}.a : \mathsf{int} \to \mathsf{char}$

型割り当ての規則を繰り返し有限回使い，型を割り当てることを,（型の）**導出**と呼び，以下のように書く．ここでι を基底型とする．

$$\dfrac{\dfrac{\dfrac{a:\iota, f:\iota\to\iota \vdash f:\iota\to\iota \qquad a:\iota, f:\iota\to\iota \vdash a:\iota}{a:\iota, f:\iota\to\iota \vdash fa:\iota}}{a:\iota \vdash \lambda f:\iota\to\iota.fa : (\iota\to\iota)\to\iota} \qquad \dfrac{a:\iota, x:\iota \vdash x:\iota}{a:\iota \vdash \lambda x:\iota.x : \iota\to\iota}}{a:\iota \vdash (\lambda f:\iota\to\iota.fa)(\lambda x:\iota.x):\iota}$$

このようにかかれた図を**導出木**と呼ぶ．一番下の型判断を木の根とみなし，上に向かって枝分かれしており，一番上の［変数］の型判断が木の葉となっているとみると，全体が木の構造をしていることが分かる．

これら［変数］,［抽象］,［作用］の規則を有限回使って，型判断 $\Gamma \vdash M:\sigma$ が導出できたとする．型付きラムダ計算では，このときはじめて，M を**型付き**λ **項**，または単に λ **項**と呼ぶ．そして項 M は Γ のもとで型σ を持つ，または項 M に型σ が付く，という．ここで M の自由な変項と型の情報は，この型割り当て規則により，型環境の Γ から得られる．例えば上の導出ではλ 項 $(\lambda f:\iota\to\iota.fa)(\lambda x:\iota.x)$ が環境 $a:\iota$ のもとで型 ι を持つことを表している．

型のないラムダ計算のすべてのλ 項に型が付くのではないことに注意しよう．例えばλ 項 ff は型が付かない．これは f が型 $\iota\to\iota$ を持つとしても，ι を持つとしても，作用項の規則を使えないので導出ができないからである．したがって，ff を部分項にもつλ 項，例えば項 $(\lambda f.ff)(\lambda f.ff)$，にも型が付かない．

B.4　β 簡　　　約

型付きラムダ計算に対してもβ 簡約を定義する．定義 7.5 と同様な方法で型付きラムダ計算の一ステップβ 簡約を定義することもできるが，ここでは導出木の考察による証明にあうように，次のような導出規則の形を用いて定義する．

定義 B.4 (一ステップβ 簡約) 前項 M の一ステップβ 簡約は以下の規則を満たす最小の関係である．

$$\frac{}{(\lambda x{:}\sigma.M)N \to_\beta M[x:=N]} \qquad \frac{M \to_\beta M'}{\lambda x{:}\sigma.M \to_\beta \lambda x{:}\sigma.M'}$$

$$\frac{M \to_\beta M'}{MN \to_\beta M'N} \qquad \frac{N \to_\beta N'}{MN \to_\beta MN'}$$

ここでメタ言語の表現 $M[x := N]$ は，型のないラムダ計算と同様に，λ 項 M に出現する（M において）自由なすべての変項 x に λ 項 N を代入して得られる λ 項を表す．これらの規則は上段が成り立つときに下段が成り立つ，という条件を表している．たとえば最後の規則は，$N \to_\beta N'$ という一ステップ β 簡約ができるときには $MN \to_\beta MN'$ という一ステップ β 簡約ができることを表す

定義 B.5 (β 簡約) 一ステップ β 簡約 $\to_\beta$ の反射推移閉包を β 簡約といい，$\to_\beta^*$ と書く．

この簡約関係 $\to_\beta$ の定義では型について何も言及していないが，正しく型付けられている λ 項にこの β 簡約を行うと得られる λ 項も正しく型付けられており，しかも型が変わらないことが示せる．この証明には，次に述べる導出に関する帰納法を用いる．

B.5　導出に関する帰納法

型付きラムダ計算の性質に関する定理では

すべての型付き λ 項 M について性質 $P(M)$ が成り立つ

という形の命題が頻繁に出てくる．すべての型付き λ 項は型割り当て規則を用いて構成されるから，これは命題

$\Gamma \vdash M : \sigma$ が導出できたとき，$P(M)$ が成り立つ

と同値である．このような形の命題を示すには「導出に関する帰納法」というものを用いるとうまくいくことが多い．本節ではこれを説明する．

型判断 $\Gamma \vdash M : \sigma$ が導出できたとすると，これを導出する導出木が存在する．このような導出木はいつも一通りしかしかない．なぜなら,［抽象］の型割り当て規則を下から上に読んでみる．下の型判断に表れる抽象の項の部分を分解するやり方は，この規則の上の型判断のようにする以外はない．すなわち他の型割り当て規則では抽象の項を分解できない.［作用］の場合も同様である．そしてこの分解は必ず［変数］の規則で終わることになる．

したがって，型判断 $\Gamma \vdash M : \sigma$ に対してこの「一通りに決まる導出に関する帰納法」を使うことができる．型付き λ 項は，型割り当ての導出木から構成されるので，型付き λ 項に関する性質の多くは項の構造に関する帰納法ではなく，導出に関する帰納法を用いることによって示される．この証明法は，2.5 節の整礎帰納法の一具体例である．

導出木に関する「一つだけ小さい部分木」の関係 $\prec$ を次のように定義する．

$$\dfrac{\vdots}{\Gamma' \vdash M' : \sigma'} \quad \prec \quad \dfrac{\vdots \qquad \dfrac{\vdots}{\Gamma' \vdash M' : \sigma'} \qquad \vdots}{\Gamma \vdash M : \sigma}$$

ここで $\prec$ の両辺の M' の上段にある $\cdots$ の部分は省略された同じ導出を表す．D をすべての導出木の集合とすると，明らかにこの $\prec$ は D 上の関係で，極小元が「根のみの導出木」となる整礎な関係を定義している．したがってこの $\prec$ についての整礎帰納法を用いることができる．これを**導出に関する帰納法**と言う．

これを用いて次が示せる．

補題 B.1 $\Gamma, x{:}\sigma \vdash L : \tau$ と $\Gamma \vdash N : \sigma$ が導出できたとする．このとき $\Gamma \vdash L[x := N] : \tau$ が導出できる．

証明: 演習問題 2.

以後簡略化のため，単に「$\Gamma \vdash L : \tau$」と書いて,「$\Gamma \vdash L : \tau$ が導出できる」ことを表すことにする.

定理 B.1 (主簡約定理) $\Gamma \vdash M : \sigma$ かつ $M \to_\beta N$ ならば，$\Gamma \vdash N : \sigma$.

証明: $\Gamma \vdash M : \sigma$ の導出に関する帰納法で証明する.

(1) $\Gamma, x{:}\sigma \vdash x : \sigma$ の場合．β 簡約は不可能である.

(2) $\Gamma \vdash \lambda x.M : \sigma \to \tau$ の場合．これは次の導出からできている.

$$\frac{\Gamma, x : \sigma \vdash M : \tau}{\Gamma \vdash \lambda x{:}\sigma.M : \sigma \to \tau}$$

$\lambda x{:}\sigma.M \to_\beta \lambda x{:}\sigma.M'$ とすると，$\to_\beta$ の定義より $M \to_\beta M'$ となっている．ここで帰納法の仮定

$$(\Gamma, x{:}\sigma \vdash M : \tau \text{ かつ } M \to_\beta M') \;\;\Rightarrow\;\; \Gamma, x{:}\sigma \vdash M' : \tau$$

を使うと，$\Gamma, x : \sigma \vdash M' : \tau$ を得る．これに［抽象］の型割り当て規則を使うと

$$\Gamma \vdash \lambda x{:}\sigma.M' : \sigma \to \tau$$

を得る.

(3) $\Gamma \vdash MN : \tau$ の場合．$\to_\beta$ の定義より，MN の形の項の β 簡約には

(i) 項 MN 自体が β 簡約される,

(ii) 項 M の部分項が β 簡約される,

(iii) 項 N の部分項が β 簡約される,

の三つの場合が考えられる．これらそれぞれの場合を仮定して考える.

(i) $MN \equiv (\lambda x{:}\sigma.L)N \to_\beta L[x := N]$ の場合

いま次の導出がある.

$$\frac{\dfrac{\Gamma, x : \sigma \vdash L : \tau}{\Gamma \vdash \lambda x{:}\sigma.L : \sigma \to \tau} \qquad \Gamma \vdash N : \sigma}{\Gamma \vdash (\lambda x{:}\sigma.L)N : \tau}$$

よって $\Gamma, x{:}\sigma \vdash L : \tau$ と補題 B.1 より $\Gamma \vdash L[x := N] : \tau$.

(ii) $MN \to_\beta M'N$ の場合

$\to_\beta$ の定義より $M \to_\beta M'$ である．いま次の導出がある.

$$\frac{\Gamma \vdash M : \sigma \to \tau \quad \Gamma \vdash N : \sigma}{\Gamma \vdash MN : \tau}$$

ここで帰納法の仮定

$$(\Gamma \vdash M : \sigma \to \tau \text{ かつ } M \to_\beta M') \;\;\Rightarrow\;\; \Gamma \vdash M' : \sigma \to \tau$$

を使うと $\Gamma \vdash M' : \sigma \to \tau$ を得る．これに［作用］の型割り当て規則を使うと $\Gamma \vdash M'N : \tau$ を得る.

(iii) $MN \to_\beta MN'$ の場合

((3)-ii) と同様に示せる.

□

B.6 停　止　性

節 B.2 で述べたように，型付きラムダ計算においては，

$$M_1 \to_\beta M_2 \to_\beta \cdots$$

と無限に続く簡約が起らないことを示すことができる．つまり，どのようにβ 簡約を繰り返し行っても，最後には必ず正規形に辿り着く．

ここで 6 章の抽象書換え系を思い出そう．まず型付きλ 項全体の集合を

$$\Lambda^{\to} = \{M \mid \Gamma \vdash M : \sigma\}$$

と定義する．定義 B.4 において $\to_\beta$ は前項上の関係として定義したが，これを型付きλ 項 $\Lambda^{\to}$ 上に制限した関係 $\to'_\beta$ を考える．定理 B.1 より $\to_\beta$ は型を変えないことが分かっているから，$\to'_\beta$ も型を変えない．$\to'_\beta$ は $\Lambda^{\to}$ 上の関係であるから，$\lambda^{\to} = (\Lambda^{\to}, \to'_\beta)$ は抽象書換え系である．このように考えると，β 簡約によってλ 項のいかなる書き換えも必ず停止することは，抽象書換え系 $\lambda^{\to}$ が SN であることに他ならない．

以下ではこの抽象書換え系 $\lambda^{\to}$ において，項 M が SN であることを「M は停止する」という．また $\to'_\beta$ を単に $\to_\beta$ と書く．

型付きλ 項の導出に関する帰納法による証明の試み

型付きラムダ計算のβ 簡約の停止性，すなわち命題：

すべての型付きλ 項 M について，M は停止する．

を証明したい．命題の形から，型付きλ 項の導出に関する帰納法を用いることをまず思いつくが，この命題についてはうまくいかない．どこで失敗するのか，まずこれについて見てみよう．型付きλ 項の導出に関する帰納法を用いて証明を試みる．

(1)　$\Gamma, x{:}\sigma \vdash x : \sigma$ の場合．β 簡約は不可能である．

(2)　$\Gamma \vdash \lambda x.M : \sigma \to \tau$ の場合．次の導出がある．

$$\frac{\Gamma, x : \sigma \vdash M : \tau}{\Gamma \vdash \lambda x{:}\sigma.M : \sigma \to \tau}$$

帰納法の仮定より M は停止する．このときは明らかに $\lambda x{:}\sigma.M$ も停止する．

(3)　$\Gamma \vdash MN : \sigma$ の場合．次の導出がある．

$$\frac{\Gamma \vdash M : \sigma \to \tau \quad \Gamma \vdash N : \sigma}{\Gamma \vdash MN : \tau}$$

したがって帰納法の仮定より，M と N は停止する．このとき項 MN が停止することを示したい．M の形について場合分けする．

(i)　$M \equiv x$ のとき，項 xN は明らかに停止する．

(ii)　$M \equiv (\lambda x.M')$ のときは，次のようにβ 簡約できる．

$$(\lambda x.M')N \to_\beta M'[x := N]$$

ここで帰納法の仮定より項 $(\lambda x.M')$ と N はそれぞれ停止することが言えるが，$M'[x := N]$ が停止するかどうかは直ちには結論できない．したがって，ここからさらに $M'[x := N]$ に対

して証明を試みなければならない．ところが，$M'[x := N]$ の形の項は，MN を導出する導出木には現れてこないから，この項に対する証明には導出に関する帰納法の仮定が使えない．

なお，M' に変項 x が複数個含まれていた場合，項 $M'[x := N]$ は変項 x の出現の数だけ N が複製されるものになる．したがって，項 $M'[x := N]$ は MN よりも大きな項になる可能性もあり，項の大きさに関する帰納法も使えないことに注意する．

このように，型付きλ 計算の停止性は，単なる型付きλ 項の導出に関する帰納法では示せないことがわかる．

停止性の証明

本節では停止性の証明を与える．証明の方針は文献 Femke van Raamsdonk and Paula Severi, On Normalisation, CS-R9545, CWI, 1995 による．この証明は次のような特徴を持つ集合 $\mathcal{SN}$ を用いる．

$$M \in \mathcal{SN} \quad \Leftrightarrow \quad \text{前項 } M \text{ は} \beta \text{ 簡約で停止する．}$$

これを使い命題

$$M \text{ が型付き} \lambda \text{ 項である (すなわち } \Gamma \vdash M : \sigma) \quad \Rightarrow \quad M \in \mathcal{SN}$$

を示すわけである．

ポイントは，集合 $\mathcal{SN}$ を「すべての停止する前項を集めたもの」という内包的な表現を使って一度に定義するのではなく，**導出**の方法を用いて「停止するであろう前項をだんだんとつくっていく」という方法を用いることである.「停止するであろう」と述べたのは，導出を使って定義した後に，集合 $\mathcal{SN}$ の要素が確かにすべての停止する前項であることを証明するためである．停止する前項すべてを直接構成することは全く自明な問題ではないが，これが可能となるのが本証明のキーとなる点である．

さて，では前項の集合 $\mathcal{SN}$ を次の導出規則で定義しよう．

$$\frac{f \text{ は変数} \quad M_1, \ldots, M_n \in \mathcal{SN}}{f M_1 \cdots M_n \in \mathcal{SN}} \qquad \frac{M \in \mathcal{SN}}{\lambda x : \sigma . M \in \mathcal{SN}}$$

$$\frac{N \in \mathcal{SN} \quad M[x := N] N_1 \cdots N_n \in \mathcal{SN}}{(\lambda x : \sigma . M) N N_1 \cdots N_n \in \mathcal{SN}}$$

これらの規則も上段が成り立つときに下段が成り立つ，という条件を表し，これらの条件を満たす最小の集合を $\mathcal{SN}$ と定義するわけである．別の言い方をすると，上の導出規則を用いて $M \in \mathcal{SN}$ が導出できたとき，前項 M が集合 $\mathcal{SN}$ に入っていると定義する．

$M \in \mathcal{SN}$ が導出された場合にも，これを導出する導出木は必ず一通りしかない．それぞれの規則を下から上に読むと，下段に現れる形の前項を分解する規則は必ずその規則のみであるからである．これにより，型付きλ 項の導出に関する帰納法による証明のときのように，今度は $M \in \mathcal{SN}$ の導出に関する帰納法が使えることになる．

最初に，集合 $\mathcal{SN}$ は元の意図通りに次のような性質を持つことを述べる．

命題 B.1 $M \in \mathcal{SN} \quad \Leftrightarrow \quad$ 前項 M はβ 簡約で停止する．

証明: 演習問題 6.

次に各型σ と型環境 Γ に対して，次の添字付き集合を定義する．

$$\Lambda_\sigma(\Gamma) = \{M \mid \Gamma \vdash M : \sigma\}$$

さらに次の添字付き集合を定義する．

$$\mathcal{SN}_\sigma(\Gamma) = \{M \mid M \in \Lambda_\sigma(\Gamma) \cap \mathcal{SN}\}$$

ここで A と B を

$$A(\Gamma) \subseteq \Lambda_\sigma(\Gamma)$$
$$B(\Gamma) \subseteq \Lambda_\tau(\Gamma)$$

となる任意の添字付き集合とする．例えば $\mathcal{SN}_\sigma$ がそのような添字付き集合である．このとき，添字付き集合 $(A \Rrightarrow B)(\Gamma)$ を以下のように定義する．

$$(A \Rrightarrow B)(\Gamma) = \{M \in \Lambda_{\sigma\to\tau}(\Gamma) \mid \forall\Gamma' \supseteq \Gamma\ \forall N \in A(\Gamma')\ \ (MN) \in B(\Gamma')\}\,.$$

また $\Rrightarrow$ を右に強く結合と仮定する．つまり $A \Rrightarrow (B \Rrightarrow C)$ を $A \Rrightarrow B \Rrightarrow C$ と略記する．

今，簡単に次が示せる．

補題 B.2 $\mathcal{SN}_{\sigma\to\tau}(\Gamma) \supseteq (\mathcal{SN}_\sigma \Rrightarrow \mathcal{SN}_\tau)(\Gamma)$

証明: 演習問題 7.

補題 B.3 $\forall\Gamma\ B_1(\Gamma) \subseteq B_2(\Gamma)$ ならば $\forall\Gamma\ (A \Rrightarrow B_1)(\Gamma) \subseteq (A \Rrightarrow B_2)(\Gamma)$

次の補題の証明には $P \in \mathcal{SN}$ の導出に関する帰納法を用いる．

補題 B.4 ι を基底型，P, N を次の型を持つλ 項とする

$$\Gamma \vdash N : \sigma_1 \to \cdots \to \sigma_n \to \iota$$
$$\Gamma, x : \sigma_1 \to \cdots \to \sigma_n \to \iota \vdash P : \tau$$

このとき，

$$N \in \mathcal{SN}_{\sigma_1} \Rrightarrow \cdots \Rrightarrow \mathcal{SN}_{\sigma_n} \Rrightarrow \mathcal{SN}_\iota\ (\Gamma)$$

かつ

$$P \in \mathcal{SN}_\tau(\Gamma, x : \sigma_1 \to \cdots \to \sigma_n \to \iota)$$

ならば，

$$P[x := N] \in \mathcal{SN}_\tau(\Gamma)$$

である．

証明: $N \in \mathcal{SN}_{\sigma_1} \Rrightarrow \cdots \Rrightarrow \mathcal{SN}_{\sigma_n} \Rrightarrow \mathcal{SN}_\iota\ (\Gamma)$ とする．命題

$$P \in \mathcal{SN}_\tau(\Gamma, x : \sigma_1 \to \cdots \to \sigma_n \to \iota) \quad \Rightarrow \quad P[x := N] \in \mathcal{SN}_\tau(\Gamma)$$

を証明する．これは次の命題と同値であるから

$$P \in \mathcal{SN} \quad \Rightarrow \quad (\Gamma, x{:}\sigma_1{\to}\cdots{\to}\sigma_n{\to}\iota \vdash P : \tau \quad \Rightarrow \quad P[x := N] \in \mathcal{SN}_\tau(\Gamma))$$

これを $P \in \mathcal{SN}$ の導出に関する帰納法で証明する．メタ言語上の略記として以下では $[x := N]$ をθ と書く．

まず $P \equiv fM_1 \cdots M_m$ のときを考える．P の型が τ でないときは命題は自明に成り立つので，P の型が τ のときを考える．

(1) $f \equiv x$ のとき．このとき補題 B.1 より $P\theta = N(M_1\theta)\cdots(M_m\theta)$ の型はτ である．今 N の型より，各 $M_i\theta(i = 1, \cdots, m)$ の型は σ_i で，さらに帰納法の仮定より $M_i\theta \in \mathcal{SN}_{\sigma_i}(\Gamma)$ が言える．このとき $m \leq n$ であることに注意すると補題 B.2 と補題 B.3 より

$$
\begin{aligned}
N &\in \mathcal{SN}_{\sigma_1} \Rrightarrow \cdots \Rrightarrow \mathcal{SN}_{\sigma_m} \Rrightarrow \mathcal{SN}_{\sigma_{m+1}} \Rrightarrow \cdots \Rrightarrow \mathcal{SN}_{\sigma_n} \Rrightarrow \mathcal{SN}_{\iota}\,(\Gamma) \\
&\subseteq \mathcal{SN}_{\sigma_1} \Rrightarrow \cdots \Rrightarrow \mathcal{SN}_{\sigma_m} \Rrightarrow \mathcal{SN}_{\sigma_{m+1}\to\cdots\to\sigma_n\to\iota}\,(\Gamma)
\end{aligned}
$$

すなわち $\sigma_{m+1} \to \cdots \to \sigma_n \to \iota = \tau$ であるので

$$N(M_1\theta)\cdots(M_m\theta) \in \mathcal{SN}_\tau(\Gamma)$$

を得る．

(2) $f \not\equiv x$ のときは，帰納法の仮定と補題 B.2 より直ちに結論を得る．

またそれぞれ $P \equiv \lambda x{:}\sigma.M,\ (\lambda x{:}\sigma.M)NN_1\cdots N_n$ のときも帰納法の仮定と補題 B.1 より直ちに結論を得る． □

補題 B.5 $\mathcal{SN}_{\sigma\to\tau}(\Gamma) \subseteq (\mathcal{SN}_\sigma \Rrightarrow \mathcal{SN}_\tau)(\Gamma)$

証明: $P \in \mathcal{SN}_{\sigma\to\tau}(\Gamma)$ のとき，すべての $N \in \mathcal{SN}_\sigma(\Gamma)$ について $PN \in \mathcal{SN}_\tau(\Gamma)$ を示す．PN の型が τ であることは明らかであるから，あとは $PN \in \mathcal{SN}$ を示せば良い．すなわち，命題

$$P \in \mathcal{SN} \quad\Rightarrow\quad (\Gamma \vdash P : \sigma \to \tau \quad\Rightarrow\quad \forall N \in \mathcal{SN}_\sigma(\Gamma)\ PN \in \mathcal{SN})$$

を型 σ の構造に関する帰納法，および $P \in \mathcal{SN}$ の導出に関する帰納法を用いて示す．

(1) σ が 基底型のとき．

Γ のもとで P の型が $\sigma \to \tau$ でなければ命題は自明に成り立つので，P の型が $\sigma \to \tau$ のときを考える．

(i) $P \equiv fM_1 \ldots M_n$ のとき．帰納法の仮定より各 $M_i \in \mathcal{SN}$．今 $N \in \mathcal{SN}$ であるから，$\mathcal{SN}$ の導出規則を使い

$$\frac{f\ \text{は変数}\quad M_1, \ldots, M_n, N \in \mathcal{SN}}{fM_1\cdots M_nN \in \mathcal{SN}}$$

(ii) $P \equiv \lambda x{:}\sigma.M$ のとき．帰納法の仮定より $M \in \mathcal{SN}_\tau(\Gamma, x{:}\sigma)$．$N \in \mathcal{SN}_\sigma(\Gamma)$ とする．σ は基底型であるから，補題 B.4 を適用して

$$M[x := N] \in \mathcal{SN}_\tau(\Gamma).$$

さらに $\mathcal{SN}$ の導出規則を使い

$$\frac{N \in \mathcal{SN}\quad M[x := N] \in \mathcal{SN}}{(\lambda x : \sigma.M)N \in \mathcal{SN}}$$

(iii) $P \equiv (\lambda x : \sigma.M)N_0N_1\cdots N_n$ のとき．

$$\frac{N_0 \in \mathcal{SN}\quad M[x := N_0]N_1\cdots N_n \in \mathcal{SN}}{(\lambda x : \sigma.M)N_0N_1\cdots N_n \in \mathcal{SN}}$$

と導出されるから，帰納法の仮定より $(M[x := N_0]N_1\cdots N_n)N \in \mathcal{SN}$．$\mathcal{SN}$ の導出規則を使い

$$\frac{N_0 \in \mathcal{SN}\quad M[x := N_0]N_1\cdots N_nN \in \mathcal{SN}}{(\lambda x : \sigma.M)N_0N_1\cdots N_nN \in \mathcal{SN}}$$

(2)　$\sigma = \sigma_1 \to \cdots \to \sigma_n \to \iota$ (ι は基底型) のとき．

それぞれ $P \equiv fM_1 \ldots M_n$, $(\lambda x : \sigma.M)N_0N_1 \cdots N_n$ のときは，上記 ((1)-i)((1)-iii) の証明と同じ．

$P \equiv \lambda x{:}\sigma.M$ のときは，$N \in \mathcal{SN}_{\sigma_1 \to \cdots \to \sigma_n \to \iota}(\Gamma)$ とすると，型の構造に関する帰納法の仮定を繰返し使うことより，$N \in \mathcal{SN}_{\sigma_1} \Rrightarrow \cdots \Rrightarrow \mathcal{SN}_{\sigma_n} \Rrightarrow \mathcal{SN}_{\iota}(\Gamma)$. よって補題 B.4 が適用でき，上記 ((1)-ii) と同様に示せる． □

命題 B.2 $\mathcal{SN}_{\sigma \to \tau}(\Gamma) = (\mathcal{SN}_\sigma \Rrightarrow \mathcal{SN}_\tau)(\Gamma)$

証明: 補題 B.2 と B.5 による． □

定理 B.2 すべての型付き λ 項は停止する．すなわち $M \in \Lambda_\sigma(\Gamma)$ ならば $M \in \mathcal{SN}_\sigma(\Gamma)$ である．

証明: $\Gamma \vdash M : \sigma$ の導出に関する帰納法で示す．B.6 節で指摘したように，問題なのは場合 $\Gamma \vdash MN : \tau$ のときのみである．帰納法の仮定と補題 B.2 より，$M \in \mathcal{SN}_{\sigma \to \tau}(\Gamma) = \mathcal{SN}_\sigma \Rrightarrow \mathcal{SN}_\tau(\Gamma)$ で，かつ $N \in \mathcal{SN}_\sigma(\Gamma)$. したがって $MN \in \mathcal{SN}_\tau(\Gamma)$ であるので，項 MN は停止する． □

B.7　合　流　性

本節では型付きラムダ計算の合流性を証明する．すなわち B.6 節で定義した抽象書換え系 $\lambda^{\to}$ の合流性を示す．この証明には 6 章のニューマンの補題を用いる．ニューマンの補題は，抽象書換え系が SN で弱合流性を持っていた場合に合流性を結論できるものだった．すでに $\lambda^{\to}$ が SN であることがわかっているので，あとは弱合流性を示せば，合流性が示せる．

弱合流性は次のように直接示すことができる．

命題 B.3 $\lambda^{\to} = (\Lambda^{\to}, \to_\beta)$ は弱合流性を持つ．

証明: $\Gamma \vdash L : \sigma$ とする．$L \to_\beta N_1$ かつ $L \to_\beta N_2$ となるときに N_1 と N_2 が同じ項に β 簡約できることを示せばよい．これを $\Gamma \vdash L : \sigma$ の導出に関する帰納法で示す．

(1)　L が変数の場合．β 簡約できない．

(2)　$L \equiv \lambda x.M$ の場合．$\lambda x.M_1 \leftarrow_\beta \lambda x.M \to_\beta \lambda x.M_2$ とすると，帰納法の仮定より共通の λ 項へと β 簡約できる．

(3)　$L \equiv MN$ の場合．MN の形の β 簡約は次に示す三つ場合が考えられる．

(i)　$L \equiv (\lambda x{:}\sigma.M)N$ のとき，M と N が正規形でなければ

- $(\lambda x{:}\sigma.M)N \to_\beta M[x := N]$
- $(\lambda x{:}\sigma.M)N \to_\beta (\lambda x{:}\sigma.M')N$
- $(\lambda x{:}\sigma.M)N \to_\beta (\lambda x{:}\sigma.M)N'$

と三通りに β 簡約できるので，これらそれぞれが同じ項に β 簡約できることを示せばよい．実際 (演習問題 3 を使い)

- $M[x := N] \to_\beta M'[x := N] {\to_\beta}^* M'[x := N']$
- $(\lambda x{:}\sigma.M')N \to_\beta (\lambda x{:}\sigma.M')N' \to_\beta M'[x := N']$
- $(\lambda x{:}\sigma.M)N' \to_\beta (\lambda x{:}\sigma.M')N' \to_\beta M'[x := N']$

とすべて同じ項に β 簡約できる．M か N が β 正規形であっても同様に示せる．

(ii)　$L \equiv MN$(M が $\lambda x{:}\sigma.K$ 以外) のとき，$MN \to_\beta M'N$ と $MN \to_\beta MN'$ の二通りに β 簡約できるが，$M'N \to_\beta M'N' \leftarrow_\beta MN'$ を持つ．

ゆえに $\lambda^{\to}$ は弱合流性を持つ． □

定理 B.3 $\lambda^{\rightarrow} = (\Lambda^{\rightarrow}, \rightarrow_\beta)$ は合流性を持つ.

証明: 定理 B.2, 命題 B.3 とニューマンの補題による. □

B.8 計算の能力

$\lambda^{\rightarrow}$ は合流性を持つから, UN 性も持つ. したがって, 型付きラムダ計算は必ず停止する計算体系で, しかも答えは一意的に決まるというよい性質を持っている. 一方, 計算の能力を考えると型のないラムダ計算よりも弱くなっている. 型の制限により, 型のないラムダ計算では使えたλ 項が, 型付きラムダ計算では型が付けられないため使えないということが起こる.

例えば, 不動点演算子のλ 項 Θ は型が付かないので型付きλ 項ではない. 型付きラムダ計算では再帰を表現することができないことになり, 帰納関数より小さいクラスの関数しか計算できないことが予想できる. 実際に次のような結果が知られている.

定理 B.4 型付きラムダ計算で定義可能な自然数上の関数全体のクラスは, 加算, 乗算と if $x = 0$ then y else z の形の条件分岐のみからなる関数を合成したクラスと正確に一致する[?, ?].

このクラスは明らかに帰納関数より小さいクラスである.

型付きラムダ計算に再帰を行うような演算子を付け加えれば, この欠点を乗り越えより計算の能力を高めることができる. このような体系の代表的なものとしてゲーデルの体系 T というものと PCF (Programming language for Computable Functions) という体系の二つがよく知られている.

ゲーデルの体系 T は, 自然数と真理値の基底型を持つ型付きラムダ計算に, 一般化された原始帰納の演算子を加えた体系である. この体系は強停止性を持ちつつ, すべての原始帰納関数が定義可能である. 正確には原始帰納関数より少し広い範囲のクラスに属する関数が定義できるが, 停止性を持つことから分かるように, 部分関数が定義できないためすべての帰納関数はカバーしていない.

PCF は自然数と真理値の基底型を持つ型付きラムダ計算に全ての型上の不動点演算子を加えた体系である[?]. この体系は強停止性を持たないが, すべての帰納関数が定義可能である.

演習問題

1. $\Gamma \vdash M : \sigma$ と $\Gamma \vdash M : \tau$ が導出できるとき, $\sigma = \tau$ である.
2. $\Gamma, x{:}\sigma \vdash L : \tau$ と $\Gamma \vdash N : \sigma$ が導出できたとする. このとき $\Gamma \vdash L[x := N] : \tau$ が導出できる.
3. $M \rightarrow_\beta M'$ ならば $M[x := N] \rightarrow_\beta M'[x := N]$.
4. $M[x := N] \rightarrow_\beta M'[x := N]$ で $N \rightarrow_\beta N'$ ならば $M[x := N] {\rightarrow_\beta}^* M'[x := N]$. また, $M[x := N] \rightarrow_\beta M'[x := N]$ でない例を挙げよ.
5. どのような前項 L も, 必ず次のどちらか一方の形であることを示せ.
 (i) $L \equiv \lambda x_1{:}\sigma_1 \cdots \lambda x_n{:}\sigma_n . f M_1 \cdots M_m \quad (n, m \geq 0)$
 (ii) $L \equiv \lambda x_1{:}\sigma_1 \cdots \lambda x_n{:}\sigma_n . (\lambda y{:}\sigma . N) M M_1 \cdots M_m \qquad (n, m \geq 0)$
6. 命題 「$M \in \mathcal{SN} \quad \Leftrightarrow \quad$ 前項 M はβ 簡約で停止する」を次のことを用いて示せ.
 (i) $[\Rightarrow]$: $M \in \mathcal{SN}$ の導出に関する帰納法を用いよ.
 (ii) $[\Leftarrow]$: 前項 M, N に対して, 関係 $\triangleleft$ を

$$M \triangleleft N \quad \Leftrightarrow \quad M \text{ は } N \text{ の部分項}$$

 と定義すると関係 $\triangleleft$ は前項上の整礎な関係である. さらに $\sharp M$ を

 「M を正規形までβ 簡約したときのβ 簡約の長さのうちの最大長の自然数」

 を表すものと定義する. もし M が正規形を持たない場合は $\sharp M$ の値は未定義とする. このとき

$$M \ll N \quad \Leftrightarrow \quad \sharp M \text{ と} \sharp N \text{ が双方とも定義されていて, かつ} \sharp M < \sharp N$$

と定義すると，これも整礎な関係である．さてさらに

$$M \prec N \quad \Leftrightarrow \quad M \lhd N \text{ または } M \ll N$$

と定義すると $\prec$ も整礎な関係である．$\prec$ に関する整礎帰納法を用いてこの [$\Leftarrow$] 方向の命題を示せ．この際，演習 5) の場合分けを用いよ．

7. 命題「$\mathcal{SN}_{\sigma\to\tau}(\Gamma) \supseteq (\mathcal{SN}_\sigma \Rrightarrow \mathcal{SN}_\tau)(\Gamma)$ 」を示せ．

8. B.5 節の導出木に関する帰納法の関係 $\prec$ を「$\cdots$ 」のような省略を使わずに定義したい．次の手順でこれを行え．

(i) すべての導出木の集合 D を正確に定義せよ．

(ii) D 上の関係 $\prec$ を省略を使わずに定義せよ．