C 言語における関数の分類

2006MI016 後藤梨沙
指導教員：佐々木克巳

1　はじめに

本研究では, 鹿島 [1] に沿って,C 言語における関数の分類を行うことを目的とする.C 言語における関数は, 計算可能なものと, 計算不可能なものに大きく分けることができる. 本稿では, まず計算可能な関数について, 計算可能な関数と帰納的に定義された関数の関係を示す. さらに, 計算不可能な関数について, 集合を対象として, 量化の重なり具合を用いた計算不可能性の度合いを分析し, その考えを基にした述語全体の階層構造である算術的階層を示す.

2　導入

ここで使われる C 言語は以下のものである.
1) データ型は整数 (int) である.
2) 計算の制御機能は, 条件分岐 (if 文), 繰返し (while 文), 関数呼び出し, および loop 文である.

定義 2.1(部分関数と定義域)
N^k の部分集合から N への対応表で規定される関数のことを, 自然数上の k 変数部分関数といい, その対応表の基になっている部分集合のことを定義域という.

定義 2.2(計算可能)
C 言語において, 自然数上の k 変数部分関数 f が計算可能であるとは, その関数を計算する k 入力プログラムが存在するということである.

3　原始帰納的関数と while 文のないプログラムとの同値性

次の定義 3.1 の関数から, 関数合成, 原始帰納法で得られる関数を原始帰納的関数という. 原始帰納的関数と C 言語の関係を定理 3.2 に示す.

定義 3.1(zero,suc,proj)
0 変数全域関数 zero,1 変数全域関数 suc, および $1 \leq i \leq k$ なる各 i, k に対する k 変数全域関数 proj_i^k を次で定義する.

$$\mathrm{zero}()=0$$
$$\mathrm{suc}(x)=x+1$$
$$\mathrm{proj}_i^k(x_1, x_2, \cdots, x_k) = x_i$$

定理 3.2(原始帰納的関数と C 言語との関係)
自然数上の全域関数 f に関する次の 2 条件は同値である.
(1)f は原始帰納的である.
(2)f を計算する関数プログラムで,while 文を使用しないものがある.

本研究では, 鹿島 [1] での定理 3.2 の証明を補い, それを卒業論文にまとめた. 本稿では, その部分は省略する.

4　帰納的部分関数と計算可能関数との同値性

定義 3.1 の関数から, 関数合成, 原始帰納法, 最小化で得られる関数を帰納的部分関数という. 帰納的部分関数と計算可能関数の関係をを定理 4.1 に示す.

定理 4.1(帰納的部分関数と計算可能関数との同値性)
自然数上の部分関数 f に関する次の 2 条件は同値である.
(1)f は帰納的である.
(2)f は計算可能である. すなわち f を計算する関数プログラムがある.

本研究では, 鹿島 [1] での定理 4.1 の証明を補い, それを卒業論文にまとめた. 本稿では, その部分は省略する.

定義から, 原始帰納的関数は計算可能関数であるといえるが, 原始帰納的でない計算可能関数は存在する. 定理 3.2 と定理 4.1 の違いとして, 条件 (1) の最小化の有無と条件 (2) の while 文の有無が挙げられる. これらのことから,while 文の使用を制限すると, それを計算するプログラムが存在しなくなる関数が存在するといえる. そのため, 最小化と while 文の機能である「一回実行する度に条件をチェックして, 続けるか否かを決める繰り返し」は, 計算可能性を表現するために必要であることがわかる. つまり, 計算可能関数は, それを計算するプログラムに while 文が使用されていたり, その関数が最小化で得られていれば, 単に計算可能関数であるが, そうでなければ原始帰納的関数にも分類されるといえる.

5　量化の重なり具合による計算不可能性の度合いの比較

定義 5.1(Σ_n 述語,Π_n 述語)
n は 1 以上の自然数,k は自然数,$\vec{x} = (x_1, x_2, \cdots, x_k)$,$\vec{y} = (y_1, y_2, \cdots, y_k)$,$A$ は k 変数述語とする.

存在量化と全称量化が交互に現れる n 個の量化 (存在量化から始まる) と $(k+n)$ 変数計算可能述語 B を用いて,A が,

$$A(\vec{x}) \Leftrightarrow (\exists y_1 \in N)(\forall y_2 \in N) \cdots B(\vec{x}, \vec{y})$$

の形で表されるとき,A は Σ_n 述語であるという.

また, 全称量化から始まる n 個の量化を用いて, 同様に Π_n 述語を定義する. 以後,「$\in N$」は省略する.

定義 5.2(Σ_n 集合,Π_n 集合)
自然数の集合 α に対して,

$$A(x) \Leftrightarrow x \in \alpha$$

となる 1 変数述語 A が Σ_n のとき,α は Σ_n 集合であるといい,A が Π_n のとき,α は Π_n 集合であるという.

定理 5.3
Σ_n 述語に否定をつけたものは,Π_n 述語であり,Π_n 述語に否定をつけたものは,Σ_n 述語である.

定理 5.4
述語 A に関する次の 2 条件は同値である.
(1)A は計算可能である.
(2)A は Σ_1 でもあり,Π_1 でもある.

鹿島 [1] の証明を補い, 証明を行う.

定理 5.4 の証明

(1) $\Rightarrow$ (2) を示す.A が 1 変数述語の場合は,

$$\alpha = \{x | A(x)\}$$

なる α が存在し,

$$A(x) \Leftrightarrow x \in \alpha$$

となる.A が計算可能であることから, 計算可能述語 B を用いて,

$A(x) \Leftrightarrow x \in \alpha$
$\quad \Leftrightarrow (\exists y)B(x, y)$

と表すことができ, また, 計算可能述語 C を用いて,

$A(x) \Leftrightarrow x \notin \bar{\alpha}$
$\quad \Leftrightarrow \neg(x \in \bar{\alpha})$
$\quad \Leftrightarrow \neg(\exists y)(C(x, y))$
$\quad \Leftrightarrow (\forall y)(\neg C(x, y))$

と表すことができる (本稿では, この詳細を省略する). よって,A は Σ_1 であり,$\neg C$ は計算可能なので,A は Π_1 でもある.A が 2 変数以上の場合も同様に示すことができる.

(2) $\Rightarrow$ (1) を示す.A が Σ_1 であり Π_1 でもあるので, 定義 5.1 から A は計算可能述語 B を用いて,

$$A(x) \Leftrightarrow \exists x B(x, \vec{y}) \qquad ①$$

と表すことができる. また,A は計算可能述語 C を用いて,

$$A(x) \Leftrightarrow \forall x C(x, \vec{y}) \qquad ②$$

と表すことができる.②より,

$$\neg A(x) \Leftrightarrow \exists x \neg C(x, \vec{y}) \qquad ②'$$

である.①, ②$'$ より,$B(1, \vec{y}), C(1, \vec{y}), B(2, \vec{y}), C(2, \vec{y}), \cdots$を計算し,$x$ に対して,$B(x, \vec{y})$ が正しければ,$A(x)$ は正しい. また $\neg C(x, \vec{y})$ が正しい, すなわち,$C(x, \vec{y})$ が正しくなければ,$A(x)$ は正しくない. このとき,$A(x)$ と $\neg A(x)$ のどちらかが正しいので, この操作は有限回で終わる. したがって,A は計算可能である. よって題意は示された. $\dashv$

定理 5.5

Σ_n 述語同士 (Π_n 述語同士) を論理積や論理和でつなげたものは, また Σ_n 述語 (Π_n 述語) である.

定理 5.6

任意の述語 A に関して次が成り立つ.

(1)A が Σ_n ならば A は Σ_{n+1} でも Π_{n+1} でもある.
(2)A が Π_n ならば A は Σ_{n+1} でも Π_{n+1} でもある.

定理 5.3,5.5,5.6 の証明は, 本稿では省略する.

定理 5.6 から述語全体の中に存在する階層構造を図 1 のように書くことができる. これを**算術的階層**といい, この図で上方に位置する述語が下方のものよりも計算不可能性の度合いが高いと解釈できる. また, この階層の各領域は空ではない.

6 算術的階層は潰れない

定理 6.1(算術的階層は潰れない)

n,k を 1 以上の任意の自然数としたとき, 次の (1),(2),(3) が成り立つ.

(1)Σ_n であって Π_n でない k 変数述語が存在する. つまり, 算術的階層の領域②,⑤,⑧,$\cdots$に k 変数述語が存在する.
(2)Π_n であって Σ_n でない k 変数述語が存在する. つまり, 算術的階層の領域③,⑥,⑨,$\cdots$に k 変数述語が存在する.
(3)Σ_{n+1} かつ Π_{n+1} であるが,Σ_n でもなく Π_n でもない k 変数述語が存在する. つまり, 算術的階層の領域④,⑦,$\cdots$に k 変数述語が存在する.

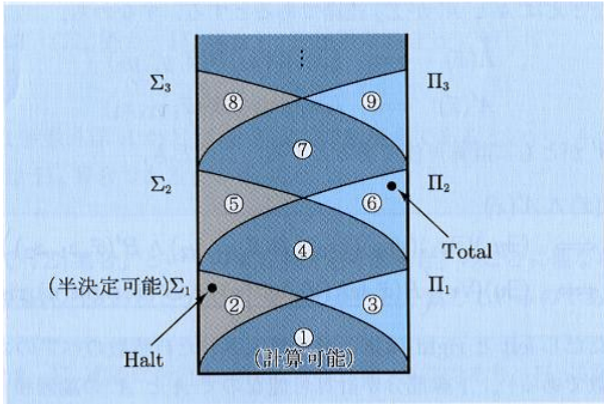

図 1:算術的階層 (鹿島 [1] より)

鹿島 [1] の証明を補い, 証明を行う.

定理 6.1 の証明

(1),(2) の証明は, 本稿では省略する.(3) を示す.(1),(2) の U と V を使って k 変数述語 W を次のように定義する. ただし,「$\dot{\div}2$」は「2 で割って余りを切り捨てた自然数」を表し,$\vec{x} = (x_1, x_2, \cdots, x_k), \vec{x'} = (x_1, x_2, \cdots, x_{k-1})$ とする.

$W(\vec{x}) \Leftrightarrow$
$\quad ((x_k\text{は偶数}) \wedge U(\vec{x'}, \frac{x_k}{2})) \vee ((x_k\text{は奇数}) \wedge V(\vec{x'}, \frac{x_k - 1}{2}))$

すると, この W が求める述語になっている.W は Σ_{n+1} かつ Π_{n+1} である. なぜなら,U は Σ_n 述語である. さらに,「x_k は偶数」は,Σ_1 述語であり, 定理 5.6 より,Σ_n でもあることになる. よって定理 5.5 より $((x_k\text{は偶数}) \wedge U(\vec{x'}, \frac{x_k}{2}))$ は Σ_n 述語である. また, 同様にして,$((x_k\text{は奇数}) \wedge V(\vec{x'}, \frac{x_k-1}{2}))$ は Π_n 述語である. ここに定理 5.6 を使うことで Σ_n,Π_n はともに,Σ_{n+1} である. よって, 定理 5.5 より W は Σ_{n+1} である.Π_{n+1} であることも同様に示すことができる. 一方,W の定義から,

$U(\vec{x}) \Leftrightarrow W(x_1, x_2, \cdots, x_{k-1}, 2x_k)$
$\quad \Leftrightarrow W(\vec{x'}, 2x_k)$
$V(\vec{x}) \Leftrightarrow W(x_1, x_2, \cdots, x_{k-1}, 2x_k + 1)$
$\quad \Leftrightarrow W(\vec{x'}, 2x_k + 1)$

が成り立つ. これより,W が Σ_n であるとすると,V も Σ_n となり (2) に反し,W が Π_n であるとすると,U も Π_n になり (1) に反するので,W は Σ_n でも Π_n でもない. よって題意は示された. $\dashv$

7 おわりに

今回,C 言語における関数の分類として, 計算可能な関数については, それに含まれる関数との関係を示し, プログラムに while 文を使用していることが計算可能性を表現するために必要な条件であることがわかった. また, 計算不可能なものについては, 量化の重なりの具合によって, 算術的階層の形に分類できることがわかった. しかし, これらのものは他の分類方法もある. そのため, これからの研究として計算可能関数をチューリング機械における計算にかかる時間による分類や, 関数そのものを C 言語以外の枠で分類していきたいと思う.

参考文献

[1] 鹿島 亮:C 言語による計算の理論. サイエンス社 (2008).