

『離散構造』 6章の例題の解答例

亀山

問1 (集合の帰納的定義) $\{L, R, 0, 1, +\}$ 上の文字列を考える. ただし, L は左括弧“(“をあらわし, R は右括弧“)”を表す. この文字列の集合 D を, 以下のように帰納的に定義する.

(D1) $0 \in D$.

(D2) $1 \in D$.

(D3) $s \in D \Rightarrow LsR \in D$.

(D4) $s \in D \wedge t \in D \Rightarrow s+t \in D$.

(a) 文字列 $0+1+0$ が, 集合 D の要素であることを示してみよう. まず, (D1) より $0 \in D$ である. また, (D2) より $1 \in D$ である. これらと, (D4) より, $0+1 \in D$ である. さらに, (D4) を使って, $0+1+0 \in D$ である.

(b) 文字列 $0+1+0$ が集合 D の要素であることを示す方法は, 上記の手順以外にもある. それを示せ. (このように2通り以上の方法で同じ文字列が導出できる時, 曖昧な文法であるという.)

(答) まず, (D2) より $1 \in D$ である. また, (D1) より $0 \in D$ である. これらと, (D4) より, $1+0 \in D$ である. さらに, (D4) を使って, $0+1+0 \in D$ である.

(c) 文字列 $LL1RR$ が, 集合 D の要素であるかどうか, 理由をつけて示せ.

(答) YES. まず, (D2) より $1 \in D$ である. また, (D3) より $L1R \in D$ である. さらに, (D3) を使って, $LL1RR \in D$ である.

(d) 文字列 $L0RR+LL1R$ が, 集合 D の要素であるかどうか, 理由をつけて示せ.

(インフォーマルな答) NO. 直感的な理由としては「かっこが整合しない」からである. (L は開き括弧, R は閉じ括弧であることに注意.)

ただ, このことを厳密に示すのは, かなり大変である. つまり, 「どういう風に導出しても, $L0RR+LL1R$ が, 集合 D の要素であることを導けない」ということを示す必要があるからである. 「導出方法」は無限にたくさんあるので, これは, 単純な話ではない.

(きちんとした証明: これは, 難しいので, 今回の授業で完全に理解する必要はない.)

文字列 $L0RR+LL1R$ が, 集合 D の要素であることを導出できたと仮定する. (ここから矛盾を導くのが目標である.) その導出の最後に使った規則は, (D3) か (D4) である.

Case-1 (最後に D3 を使った場合) (D3) を使う直前では, $0RR+LL1$ が D の要素であることが導けているはずである. これの導出の最後は, (D4) しかあり得ない. そして, (D4) の直前は, $0RR$ と $LL1$ が D の要素であることが導けていたはずである. しかし, $0RR$ を導くことはどうやってもできない. (その導出の最後が, (D1),(D2),(D3),(D4) のどれだとしても矛盾する.) よって矛盾する.

Case-2 (最後に D4 を使った場合) (D4) を使う直前では, $L0RR$ と $LL1R$ が, 集合 D の要素であることを導出できたはずである. $L0RR$ に着目すると, それが D の要素であることを示したときの最後に使ったルールは, (D3) のはずである. そして (D3) の直前は, $0R$ が D の要素であることがわかっていたはずである. しかし, (D1),(D2),(D3),(D4) のどれを使っても $0R \in D$ は導けない. よって矛盾である.

以上により, どのケースでも矛盾したので, 最終的に, $L0RR+LL1R$ が集合 D の要素であるという仮定が間違いであったことがわかる. (証明終わり)

(e) この文法を曖昧でないものにする (D に属するどの文字列も、唯一つの方法で導けるようにする) には、 D の定義をどう変更したらよいか?

(答) この解答は1つではない。一例を以下に示す。

(D1) $0 \in D$.

(D2) $1 \in D$.

(D3) $s \in D \Rightarrow LsR \in D$.

(D4) $s \in D \Rightarrow s + 0 \in D$.

(D5) $s \in D \Rightarrow s + 1 \in D$.

(D6) $s \in D \wedge t \in D \Rightarrow s + LtR \in D$.

このやりかたは、 $s + t$ の構成のときの t として、0 か 1 か LuR の形のものを許す ($v + w$ の形は許さない) というアイデアに基づいている。この場合、+ の記号は、左側が優先される。たとえば、 $0 + 0 + 0$ は $(0 + 0) + 0$ の意味になる。このようなとき、「+ が左結合的 (left associative)」という。

問2 (関数の帰納的定義) リストを反転する関数 $\text{reverse}: List_A \rightarrow List_A$ の定義を与えよ。たとえば、 $\text{reverse}(\langle 1, 2, 3 \rangle) = \langle 3, 2, 1 \rangle$ となる。

(答)

$$\text{reverse}(L) = \begin{cases} \langle \rangle & \text{if } L = \langle \rangle \\ \text{reverse}(L') \oplus \langle x \rangle & \text{if } L = \text{cons}(x, L') \end{cases}$$

ここで、 $\text{reverse}(L') \oplus \langle x \rangle$ で使っている \oplus は2つのリストを接続する関数 (append 関数) である。また、 $\text{reverse}(L') \oplus \langle x \rangle$ を $\text{reverse}(L') \oplus x$ としてはいけないし、 $\text{reverse}(L') :: x$ としてもいけない。(これらは、いずれも、リストが来るべき場所に、要素が来てしまっている。)

(計算例)

$$\begin{aligned} \text{reverse}(\langle 1, 2 \rangle) &= \text{reverse}(\langle 2 \rangle) \oplus \langle 1 \rangle \\ &= (\text{reverse}(\langle \rangle) \oplus \langle 2 \rangle) \oplus \langle 1 \rangle \\ &= (\langle \rangle \oplus \langle 2 \rangle) \oplus \langle 1 \rangle \\ &= \langle 2 \rangle \oplus \langle 1 \rangle \\ &= \langle 2, 1 \rangle \end{aligned}$$

問3 (リストに関する帰納法) 任意のリスト x, y, z に対して、

$$(x \oplus y) \oplus z = x \oplus (y \oplus z)$$

が成立することを x に関する帰納法を用いて証明せよ。ただし、 \oplus はリストを連結する関数である。

(答) 上記の式を (式1) と呼び、これをリスト x に関する帰納法で証明する。

(Case: $x = \langle \rangle$ のとき)

(式1) の左辺は、 $(\langle \rangle \oplus y) \oplus z = y \oplus z$ となる (\oplus の定義により。) 一方、(式1) の右辺は、 $\langle \rangle \oplus (y \oplus z) = y \oplus z$ となる。 (\oplus の定義により。) よって、両辺は等しく、(式1) が成立する。

(Case: $x = \text{cons}(u, L)$ のとき)

$x = L$ に対して、式1 が成立すると仮定して、 $x = \text{cons}(u, L)$ に対して、式1 が成立することを証明したい。

仮定を書きなおすと、

$$(L \oplus y) \oplus z = L \oplus (y \oplus z)$$

である。

$$\begin{aligned}x = \text{cons}(u, L) \text{ の時の式 1 の左辺} &= (\text{cons}(u, L) \oplus y) \oplus z \\ &= \text{cons}(u, (L \oplus y)) \oplus z \\ &= \text{cons}(u, (L \oplus y) \oplus z)\end{aligned}$$

$$\begin{aligned}x = \text{cons}(u, L) \text{ の時の式 1 の右辺} &= \text{cons}(u, L) \oplus (y \oplus z) \\ &= \text{cons}(u, (L \oplus (y \oplus z)))\end{aligned}$$

上記の「仮定」をつかうと、左右両辺は一致していることがわかる。

よって、 $x = L$ に対して、式 1 が成立すると仮定して、 $x = \text{cons}(u, L)$ に対して、式 1 が成立することを証明できた。

以上の 2 つの Case よりリストに関する帰納法をつかって、(式 1) が任意のリスト x に対して成立することがいえた。(証明終わり)