『離散構造』 6 章 (帰納) の演習問題の解答
2010.2.19, 亀山

問題 1 (集合の帰納的定義)

(a) $\Sigma = \{a, b, c\}$ とするとき，Σ 上の文字列で，前から読んでも後ろから読んでも同じ文字列となるものの集合 A を帰納的に定義せよ．ただし，空文字列も含むとする．

解答: この種の問題は、解答は 1 つとは限らない。以下に一例をあげる。

- $\Lambda \in A$.
- $x \in \Sigma \Rightarrow x \in A$.
- $(x \in \Sigma \wedge s \in A) \Rightarrow xsx \in A$.

この問題で、ひっかかりやすいのは、$\Lambda \in A$ と、$x \in A$ の両方が必要である点である。普通は、どちらかだけを「たね」にして、帰納的定義を構築できるが、本問の場合は、空文字列から始めて第三の定義節を使っていくと偶数長の文字列が構成され、長さ 1 の文字列から始めて第三の定義節を使っていくと奇数長の文字列が構成されるので、両方とも「たね」として必要である。

(b) 自然数のリスト (要素が全て自然数であるリスト) の中で，要素が昇順に並んでいるものの集合を帰納的に定義せよ．ただし，空リストも含むとする．

解答の一例: 定義したい集合を L とする。

- $\langle\rangle \in L$.
- $n \in \mathcal{N} \Rightarrow \langle n \rangle \in L$.
- $(n \in \mathcal{N} \wedge l \neq \langle\rangle \wedge \mathrm{head}(l) \geq n) \Rightarrow cons(n, l) \in L$.

この定義でも、1 番目と 2 番目の両方が必要 (長さ 0 のリストと、長さ 1 のリストの両方が「たね」として必要) なところが、間違いやすいところである。

問題 2 (帰納的定義に関する導出) 次の帰納的定義により、p 式の集合 P を定める．(本問では，定義される p 式を，地の文と区別するため「..」で囲うことにする.「」そのものは p 式の一部ではないことに注意せよ．)

- 「1」,「2」, $\cdots$,「10」 は，p 式である．
- 「e_1」と「e_2」 が p 式ならば,「e_1 e_2 +」 は p 式である．
- 「e_1」と「e_2」 が p 式ならば,「e_1 e_2 −」 は p 式である．

(a) 「1 2 3 + −」 が p 式であることを示せ．

答. 最初なので、最高に丁寧に記述する。

a. 「2」は p 式である。 (定義の 1 番目より)
b. 「3」は p 式である。 (定義の 1 番目より)
c. 「2 3 +」は p 式である。(上記の a,b と定義の 2 番目より)
d. 「1」は p 式である。(定義の 1 番目より)
e. 「1 2 3 + −」は p 式である。(上記の c,d と定義の 3 番目より)

(b) 「1 2 3 + − +」 が p 式であるかどうか，理由とともに示せ．

答. p 式でない。

理由についての前振りの説明: 一般に「帰納的定義された集合に○○という要素が属している」こと示すのは簡単だが、「。。。属していない」方を厳密に示すのはかなり難しい。この問題も、1 年生の授業

の範囲を越えていると思う。(←そんな問題を出すな、と言われそうだが、やっぱり、これがわかってこそ、帰納的定義の神髄がわかる、とも言える大事なポイントなので、演習で出題せずにはいられなかった。) したがって、ここでの説明は完全に理解できなくてもよいし、期末試験に出すこともない。

演習において解答してくれた人は、「p 式では、1,2,3 などの定数の個数と、$+, -$ という演算子の個数とは、常に前者が 1 つ多いはずだが、この式は同数なので、p 式でない。」という説明をしてくれた。これは、素晴しい!

ただし、この「。。。常に前者が 1 つ多いはず」ということを証明しなければいけないので、ここで、その証明を与えよう。

「p 式 e においては、e に含まれる定数の個数から、e に含まれる演算子の個数を引けば 1 である」ことを、p 式に関する帰納法で証明する。

- まず、e が定数であれば、明らかに成立する。
- $e = e_1\ e_2\ +$ であったとする。この場合、使ってよい「帰納法の仮定」は、e_1 と e_2 に対して、それぞれ目的とする性質 (差が 1 である)」である。そこで、e における定数の個数から演算子の個数を引けば、$e = e_1\ e_2\ +$ なので、やはり、差が 1 である。
- $e = e_1\ e_2\ -$ の場合も同様。

以上から、すべての p 式 e においては、定数の個数から演算子の個数を引けば 1 であることが証明できた。

(c) 「1 2 3 4 5 + − + −」 が p 式であるかどうか，理由とともに示せ.

答. p 式である。

a. 「4」は p 式である。 (定義の 1 番目より)
b. 「5」は p 式である。 (定義の 1 番目より)
c. 「4 5 +」は p 式である。(上記の a,b と定義の 2 番目より)
d. 「3」は p 式である。(定義の 1 番目より)
e. 「3 4 5 + −」は p 式である。(上記の c,d と定義の 3 番目より)
f. 「2」は p 式である。(定義の 1 番目より)
g. 「2 3 4 5 + − +」は p 式である。(上記の e,f と定義の 3 番目より)
h. 「1」は p 式である。(定義の 1 番目より)
i. 「1 2 3 4 5 + − +」は p 式である。(上記の g,h と定義の 3 番目より)

(d) p 式の文法には曖昧さがない．すなわち，どんな p 式に対しても，それが p 式であることを示す導出は 1 通りしかない．このことの理由を示せ.

答. (本問も、言うまでもなく、1 年生の授業の範囲を越えている。近い将来には、是非これを理解してもらいたいと思っているが、今回の授業では、理解できなくてもよい。)

p 式に関する帰納法で、「導出が 1 通り」を証明する。

e を p 式とする。e が p 式であることの導出を考える。e の一番右の記号を見れば、その最後に使った定義節が 1 通りに定まる。

まず、一番右の記号が定数であったとする。すると、明らかに e は、その定数のみから構成される p 式であり、その導出は 1 通りである。

次に、$e = e_1\ e_2\ +$ とする。このとき、さらに、$e = e_3\ e_4\ +$ というようにあらわされたとすると (ここで各 e_i は p 式)、$e_1 = e_3$ かつ $e_2 = e_4$ であることが証明できる。(つまり、この分解は一意的である。この一意性の証明はややこしいので、この文書の最後に記述する。) よって、この場合、e が p 式であることを示す導出の最後の部分は、一意的である。また、e_1 と e_2 も p 式なので、(帰納法の仮定より) それらの導出も一意的である。よって、e の導出も一意的である。

$e = e_1\ e_2\ -$ の場合も、上と同様である。(証明おわり)

問題 3 (関数の帰納的定義)

前問で定義した集合 P に対して $f : P \to \mathcal{N}$ となる関数 f と $g : P \to \mathcal{Z}$ となる関数 g を以下のように定義する.

$$f(e) = \begin{cases} 1 & \text{if } e = 「1」, 「2」, \cdots, 「10」 \\ f(e_1) + f(e_2) + 1 & \text{if } e = 「e_1\ e_2\ +」 \\ f(e_1) + f(e_2) + 1 & \text{if } e = 「e_1\ e_2\ -」 \end{cases}$$

$$g(e) = \begin{cases} e & \text{if } e = 「1」, 「2」, \cdots, 「10」 \\ g(e_1) + g(e_2) & \text{if } e = 「e_1\ e_2\ +」 \\ g(e_1) - g(e_2) & \text{if } e = 「e_1\ e_2\ -」 \end{cases}$$

これに対して，以下の問に答えよ.

(a) $f(3\ 2\ 1\ +\ -)$ の値を求めよ.

$f(3\ 2\ 1\ +\ -) = f(3) + f(2\ 1\ +) + 1 = 1 + (f(2) + f(1) + 1) + 1 = 1 + (1 + 1 + 1) + 1 = 5$ である。

(b) $g(3\ 2\ 1\ +\ -)$ の値を求めよ.

$g(3\ 2\ 1\ +\ -) = g(3) - g(2\ 1\ +) = 3 - (g(2) + g(1)) = 3 - (2 + 1) = 0$ である。

[補足] 関数 g の定義の書きかたが、わかりにくかったようで、$g(3) = e$ だと思ってしまった人がいたようだ。実は、$g(3)$ と書くときの 3 は「3 という文字列」なのに対して、$g(3)$ の値の方は「自然数としての 3」のつもりであり、それをごっちゃに書いたので、わからなかった人がいた。これは、皆さんのせいではなく、問題の書き方の方が悪かったせいであり、(Computer Science では、「自然数の 3」と「文字列としての “3”」を区別するのは基本中の基本である)、皆さんとしては気にしなくてよい。

上記のような誤解をなくすためには、本当は、$g($「3」$) = 3$ という風に、「文字列としての 3 の方は、「」で囲って書くべきであった。

(c) 任意の p 式 e に対して，$f(e)$ が奇数であることを，p 式に関する帰納法を使って証明せよ.

これは、是非とも習得してほしい、帰納法の基本形である。

- $e = 1, 2, \cdots, 10$ のとき、$f(e) = 1$ であり、これは奇数なので、上記の命題は成立する。
- $e = e_1\ e_2\ +$ のとき、$f(e) = f(e_1) + f(e_2) + 1$ である。帰納法の仮定 (この場合に使ってよい命題) は、「$f(e_1)$ が奇数」と「$f(e_2)$ が奇数」の 2 つである。これらから、$f(e)$ が奇数であることが言える。
- $e = e_1\ e_2\ -$ のとき、これは、1 つ上のケースとまったく同様である。

以上から、p 式に関する帰納法により、「任意の p 式 e に対して、$f(e)$ が奇数である」ことが言えた。(証明おわり)

(d) 任意の p 式 e に対して，$|g(e)| \leq 5f(e) + 5$ であることを，p 式に関する帰納法を使って証明せよ.

これも帰納法の基本形である。

- $e = 1, 2, \cdots, 10$ のとき、$f(e) = 1$ であり、$g(e) = e$ である。よって、$|g(e)| \leq 10 = 5f(e) + 5$ となって、上記の命題は成立する。
- $e = e_1\ e_2\ +$ のとき、$f(e) = f(e_1) + f(e_2) + 1$ であり、$g(e) = g(e_1) + g(e_2)$ である。帰納法の仮定 (この場合に使ってよい命題) は、「$|g(e_1)| \leq 5f(e_1) + 5$」と「$|g(e_2)| \leq 5f(e_2) + 5$」である。そこで、

$$|g(e)| = |g(e_1) + g(e_2)| \leq |g(e_1)| + |g(e_2)| \leq (5f(e_1) + 5) + (5f(e_2) + 5) \leq 5f(e) + 5$$

となり、上記の命題は e に対しても成立する。

- $e = e_1\ e_2\ -$ のとき、$f(e) = f(e_1) + f(e_2) + 1$ であり、$g(e) = g(e_1) - g(e_2)$ である。帰納法の仮定 (この場合に使ってよい命題) は、「$|g(e_1)| \leq 5f(e_1) + 5$」と「$|g(e_2)| \leq 5f(e_2) + 5$」である。そこで、

$$|g(e)| = |g(e_1) - g(e_2)| \leq |g(e_1)| + |g(e_2)| \leq (5f(e_1) + 5) + (5f(e_2) + 5) \leq 5f(e) + 5$$

となり、上記の命題は e に対しても成立する。

以上から、p 式に関する帰納法により、「任意の p 式 e に対して、$|g(e)| \leq 5f(e) + 5$ である」ことが言えた。(証明おわり)

補足

問題 2-(d) の証明の補足:

p 式 e が $e = e_1\ e_2\ + = e_3\ e_4\ +$ と分解され、e_1, e_2, e_3, e_4 が p 式であったとすると、$e_1 = e_3$ かつ $e_2 = e_4$ であることを証明する。(つまり、2 通りに分解されることはない。)

まず、p 式 e, e_1, e_2, e_3, e_4 に対して、$e = e_1\ e_2\ + = e_3\ e_4\ +$ と仮定する。

Case-1: e_1 の方が e_3 より (文字列として) 長いとする。この場合、ある文字列 s に対して、$e_1 = e_3\ s$ となる。また、$s\ e_2 = e_4$ である。

ここで、問題 2-(b) において証明したように、どんな p 式 e に対しても、それにあらわれる定数 $1, 2, \cdots, 10$ の個数から、演算子 $+, -$ の個数を引くと、1 である。したがって、$s\ e_2 = e_4$ ということから、s においては、定数の個数と演算子の個数は等しい。

しかし、これは矛盾である。なぜなら、p 式 e_4 の先頭から 1 文字以上とった式 s において、定数の個数と演算子の個数が等しくなることはない (先頭から何文字取ろうと、常に、定数の個数の方が演算子の個数より多い) からである。[厳密には、このことも p 式に関する帰納法で証明する必要がある。]

Case-2: e_1 の方が e_3 より (文字列として) 短いとする。これは Case-1 と同様に矛盾する。

よって、e_1 と e_3 は一致する場合しか残らず、その場合、e_2 と e_4 も一致する。(証明おわり。)