

岡本 吉央
 okamotoy@uec.ac.jp

電気通信大学

2013 年 7 月 23 日

最終更新 : 2013 年 7 月 22 日 15:06

目次

① 関数の冪乗

② 集合の再帰的定義

③ 今日のまとめ

関数の冪乗 : 例題

例題 1

関数 $f: \mathbb{R} \rightarrow \mathbb{R}$ を

$$f(x) = 2x^2$$

と定義する. このとき, 任意の正の整数 n に対して

$$f^n(x) = 2^{2^n-1}x^{2^n}$$

が成り立つことを証明せよ

確認

- ▶ $n = 2$ のとき : $f^2(x) = f(2x^2) = 2(2x^2)^2 = 2^3x^4$
- ▶ $n = 3$ のとき : $f^3(x) = f(2^3x^4) = 2(2^3x^4)^2 = 2^7x^8$
- ▶ $n = 4$ のとき : $f^4(x) = f(2^7x^8) = 2(2^7x^8)^2 = 2^{15}x^{16}$
- ▶ ...

関数の冪乗 : 例題 — 証明 (2)

例題 1

関数 $f: \mathbb{R} \rightarrow \mathbb{R}$ を

$$f(x) = 2x^2$$

と定義する. このとき, 任意の正の整数 n に対して

$$f^n(x) = 2^{2^n-1}x^{2^n}$$

が成り立つことを証明せよ

証明 (帰納段階) : 次に, 任意の正の整数 k を考える.

- ▶ $f^k(x) = 2^{2^k-1}x^{2^k}$ が正しいと仮定する.
- ▶ 目標 : $f^{k+1}(x) = 2^{2^{k+1}-1}x^{2^{k+1}}$ を示すこと

概要

今日の目標

- ▶ 再帰的定義を通して, 関数の冪乗を理解する
- ▶ 集合を再帰的に定義する方法を理解する

今後の予定

- ▶ 7月30日(火)6限 @ A201 : 補講 (試験範囲に含まない)
- ▶ 8月6日(火)6限 @ A201 : 期末試験

期末試験は, A4 用紙一枚 (両面自筆書き込み) のみ持ち込み可

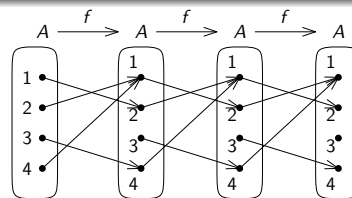
関数の冪乗

集合 A と関数 $f: A \rightarrow A$

関数の冪乗とは?

0 以上の整数 n に対して f の冪乗 $f^n: A \rightarrow A$ を次で定義する

$$f^n = \begin{cases} \text{id}_A & (n = 0 \text{ のとき}) \\ f \circ f^{n-1} & (n > 0 \text{ のとき}) \end{cases}$$



- ▶ $f(1) = 2, f(2) = 1, f(3) = 4, f(4) = 1$
- ▶ $f^2(1) = 1, f^2(2) = 2, f^2(3) = 1, f^2(4) = 2$
- ▶ $f^3(1) = 2, f^3(2) = 1, f^3(3) = 2, f^3(4) = 1$

関数の冪乗 : 例題 — 証明 (1)

例題 1

関数 $f: \mathbb{R} \rightarrow \mathbb{R}$ を

$$f(x) = 2x^2$$

と定義する. このとき, 任意の正の整数 n に対して

$$f^n(x) = 2^{2^n-1}x^{2^n}$$

が成り立つことを証明せよ

証明 (基底段階) : まず, $n = 1$ のときに正しいことを証明する.

- ▶ 左辺 = $f^1(x) = f(x) = 2x^2$
- ▶ 右辺 = $2^{2^1-1}x^{2^1} = 2x^2$
- ▶ したがって, $n = 1$ のとき, $f^n(x) = 2^{2^n-1}x^{2^n}$ は正しい.

関数の冪乗 : 例題 — 証明 (3)

$f^{k+1}(x)$

$$\begin{aligned} &= (f \circ f^k)(x) && \text{(関数の冪乗の定義)} \\ &= f(f^k(x)) && \text{(関数の合成の定義)} \\ &= f(2^{2^k-1}x^{2^k}) && \text{(帰納法の仮定)} \\ &= 2(2^{2^k-1}x^{2^k})^2 && \text{(f の定義)} \\ &= 2^{1+2(2^k-1)}x^{2 \cdot 2^k} && \text{(計算して整理)} \\ &= 2^{2^{k+1}-1}x^{2^{k+1}} && \text{(更に整理)} \end{aligned}$$

したがって, $f^{k+1}(x) = 2^{2^{k+1}-1}x^{2^{k+1}}$ は正しい. □

① 関数の冪乗

② 集合の再帰的定義

③ 今日のまとめ

文字列の定義

文字の集合 Σ

文字列とは？

Σ 上の文字列とは、次を満たすものこと

- 1 ϵ は Σ 上の文字列である (ϵ は空列を表す)
- 2 s が Σ 上の文字列であり、 $x \in \Sigma$ ならば、 xs も Σ 上の文字列である
- 3 このようにして生成されるものだけが Σ 上の文字列である

Σ 上の文字列をすべて集めた集合を Σ^* で表す

例: $\Sigma = \{a, b\}$ のとき

$$\Sigma^* = \{\epsilon, a, b, aa, ab, ba, bb, aaa, baa, aab, bab, aba, bba, abb, bbb, \dots\}$$

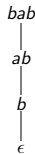
再帰的定義を理解する (2): 認識する

文字列とは？

Σ 上の文字列とは、次を満たすものこと

- 1 ϵ は Σ 上の文字列である (ϵ は空列を表す)
- 2 s が Σ 上の文字列であり、 $x \in \Sigma$ ならば、 xs も Σ 上の文字列である
- 3 このようにして生成されるものだけが Σ 上の文字列である

Σ 上の文字列をすべて集めた集合を Σ^* で表す



文字列の長さ

文字列の長さ (直感に基づく定義)

文字列 $s \in \Sigma^*$ の長さは、 s に含まれる Σ の要素の数

文字列	長さ
ϵ	0
a	1
b	1
aa	2
abb	3
$baabaabb$	8

ちゃんと定義するには？

文字列の再帰的定義に沿って、その長さも再帰的に定義する

「辞書」をどのように定義するか？

「辞書」をどのように定義するか？

▶ 辞書は単語を集めたもの $\rightsquigarrow \therefore$ 辞書は**単語の集合**

「単語」をどのように定義するか？

▶ 単語は文字を並べたもの $\rightsquigarrow \therefore$ 単語は**文字の列**

「文字」をどのように定義するか？

▶ 文字は**集合**

英語ならば、 $\{a, b, c, d, \dots, x, y, z\}$

「列」をどのように定義するか？

これがここからの話

▶ 再帰的に定義する

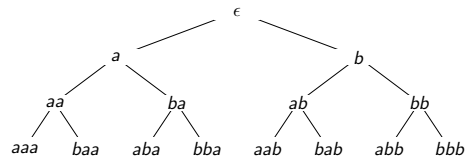
再帰的定義を理解する (1): 生成する

文字列とは？

Σ 上の文字列とは、次を満たすものこと

- 1 ϵ は Σ 上の文字列である (ϵ は空列を表す)
- 2 s が Σ 上の文字列であり、 $x \in \Sigma$ ならば、 xs も Σ 上の文字列である
- 3 このようにして生成されるものだけが Σ 上の文字列である

Σ 上の文字列をすべて集めた集合を Σ^* で表す



再帰的定義を理解する (2): 認識する

文字列とは？

Σ 上の文字列とは、次を満たすものこと

- 1 ϵ は Σ 上の文字列である (ϵ は空列を表す)
- 2 s が Σ 上の文字列であり、 $x \in \Sigma$ ならば、 xs も Σ 上の文字列である
- 3 このようにして生成されるものだけが Σ 上の文字列である

Σ 上の文字列をすべて集めた集合を Σ^* で表す



bca は $\{a, b\}$ 上の文字列ではない

文字列の長さ: 再帰的定義

文字列の長さ: 再帰的定義

文字列 $s \in \Sigma^*$ の長さ $\ell(s)$ を次のように定義する

- 1 $\ell(\epsilon) = 0$
- 2 $s \in \Sigma^*$ かつ $x \in \Sigma$ ならば、 $\ell(xs) = 1 + \ell(s)$

$$\begin{aligned} \ell(babb) &= 1 + \ell(abb) \\ &= 1 + (1 + \ell(bb)) \\ &= 1 + (1 + (1 + \ell(b))) \\ &= 1 + (1 + (1 + (1 + \ell(\epsilon)))) \\ &= 1 + (1 + (1 + (1 + 0))) \\ &= 4 \end{aligned}$$

注意

より正確には、 $\ell: \Sigma^* \rightarrow \mathbb{R}$ という関数を再帰的に定義している

文字列の性質

文字の集合 Σ

例題

 Σ 上の任意の文字列 $s \in \Sigma^*$ と任意の文字 $x \in \Sigma$ に対して

$$sx \in \Sigma^*$$

となることを証明せよ

証明の方針

 s の長さに関する帰納法 $(s$ の長さは 0 以上なので、「長さが 0」のときが基底段階)

文字列の性質：証明 (2)

証明 (帰納段階) :

- ▶ $\ell(s) = k \geq 0$ である任意の $s \in \Sigma^*$ と任意の $x \in \Sigma$ に対して, $sx \in \Sigma^*$ であると仮定する

証明の目標

 $\ell(s) = k + 1$ である任意の $s \in \Sigma^*$ と任意の $x \in \Sigma$ に対して, $sx \in \Sigma^*$

- ▶ $\ell(s) = k + 1 \geq 1$ なので, ある $t \in \Sigma^*$ とある $y \in \Sigma$ が存在して $s = yt$
- ▶ このとき, $k + 1 = \ell(s) = \ell(yt) = 1 + \ell(t)$ なので, $\ell(t) = k$
- ▶ 帰納法の仮定より, $tx \in \Sigma^*$
- ▶ 文字列の定義より, $ytx \in \Sigma^*$
- ▶ $yt = s$ なので, $sx \in \Sigma^*$ □

回文を再帰的に定義する

文字の集合 Σ

回文とは?

 Σ 上の回文とは, Σ 上の文字列で次を満たすもののこと

- 1 ϵ は Σ 上の回文である
- 2 任意の $x \in \Sigma$ に対して x は Σ 上の回文である
- 3 s が Σ 上の回文であり, $x \in \Sigma$ ならば, xsx も Σ 上の回文である
- 4 このようにして生成されるものだけが Σ 上の回文である

例: $\Sigma = \{a, b\}$ のとき $\epsilon, a, b, aa, bb, aaa, bab, aba, bbb, \dots$

例題：例を見ている

次の関数 $f: \Sigma^* \rightarrow \Sigma^*$ を考える

- 1 $f(\epsilon) = \epsilon$
- 2 $s \in \Sigma^*$ かつ $x \in \Sigma$ ならば, $f(xs) = xxf(s)$

$$\begin{aligned}
 f(abbaa) &= aaf(bbaa) \\
 &= aabbbf(baa) \\
 &= aabbbbf(aa) \\
 &= aabbbbbaaf(a) \\
 &= aabbbbbaaaf(\epsilon) \\
 &= aabbbbbaaaa
 \end{aligned}$$

文字列の性質：証明 (1)

証明 (基底段階) : $\ell(s) = 0$ のとき, すなわち, $s = \epsilon$ のとき

- ▶ $sx = x$
- ▶ $\epsilon \in \Sigma^*$ かつ $x \in \Sigma$ なので, 文字列の定義より $x \in \Sigma^*$
- ▶ したがって, $sx \in \Sigma^*$

「回文」を定義する

回文 (かいぶん)

(デジタル大辞泉)

- 1 複数の人に順に回して知らせるようにした手紙や通知。回状。まわしづみ。かいもん。
- 2 和歌・俳諧などで、上から読んでも下から逆に読んでも同じ音になるように作ってある文句。「たけやぶやけた」の類。かいもん。

2 の意味での回文の例

(http://kaibunfan.com/より)

- ▶ できたら、しらたきで。
- ▶ ごつい、ドイツ語。
- ▶ 静岡を图示。
- ▶ 寒い！タンメンタイムさ。
- ▶ 良い知らせらしいよ。
- ▶ 風邪、なぜか、なかなか長い、なかなか風邪。なぜか。

例題：次の関数はどんな操作を行う関数だろうか？

次の関数 $f: \Sigma^* \rightarrow \Sigma^*$ を考える

- 1 $f(\epsilon) = \epsilon$
- 2 $s \in \Sigma^*$ かつ $x \in \Sigma$ ならば, $f(xs) = xxf(s)$

考えたいこと

- ▶ この関数 f が行う操作は何なのか？
- ▶ この関数 f がうまく定義されているか？ ($f(s) \in \Sigma^*$ なのか？)

例題：うまく定義されていること

次の関数 $f: \Sigma^* \rightarrow \Sigma^*$ を考える

- 1 $f(\epsilon) = \epsilon$
- 2 $s \in \Sigma^*$ かつ $x \in \Sigma$ ならば, $f(xs) = xxf(s)$

証明すること

任意の $s \in \Sigma^*$ に対して $f(s) \in \Sigma^*$ であること s の長さに関する帰納法で証明する

例題：うまく定義されていること (証明)

証明 (基底段階) : $\ell(s) = 0$ のとき, すなわち, $s = \epsilon$ のとき

- ▶ f の定義より $f(\epsilon) = \epsilon$ なので, 文字列の定義より $f(\epsilon) = \epsilon \in \Sigma^*$

証明 (帰納段階) :

- ▶ $\ell(s) = k \geq 0$ である任意の $s \in \Sigma^*$ に対して, $f(s) \in \Sigma^*$ であると仮定する

証明の目標

$\ell(s) = k + 1$ である任意の $s \in \Sigma^*$ に対して, $f(s) \in \Sigma^*$

- ▶ $\ell(s) = k + 1 \geq 1$ なので, ある $t \in \Sigma^*$ とある $x \in \Sigma$ が存在して $s = xt$
- ▶ このとき, $k + 1 = \ell(s) = \ell(xt) = 1 + \ell(t)$ なので, $\ell(t) = k$
- ▶ 帰納法の仮定より, $f(t) \in \Sigma^*$
- ▶ 文字列の定義より, $xf(t) \in \Sigma^*$ となり, さらに $xxf(t) \in \Sigma^*$
- ▶ f の定義より $f(s) = f(xt) = xxf(t)$ なので, $f(s) = xxf(t) \in \Sigma^*$ □

目次

- 1 関数の冪乗
- 2 集合の再帰的定義
- 3 今日のまとめ

今日のまとめ

今日のまとめ

今日の目標

- ▶ 再帰的定義を通して, 関数の冪乗を理解する
- ▶ 集合を再帰的に定義する方法を理解する

今後の予定

- ▶ 7月30日 (火) 6限 @ A201 : 補講 (試験範囲に含まない)
- ▶ 8月6日 (火) 6限 @ A201 : 期末試験

期末試験は, A4 用紙一枚 (両面自筆書き込み) のみ持ち込み可