

12 再帰的プログラム

離散数学第12回講義

離散数学12

3

12.1 関数の再帰的定義 (recursive definition)

階乗関数 $f(n) = n!$ の再帰性

$$f(n) = n! = n \times (n-1) \times (n-2) \times \cdots \times 2 \times 1$$

$$(n-1)! = f(n-1)$$

$$(I) \quad f(0) = 1$$

$$(II) \quad f(n) = n \times f(n-1)$$

$$\begin{aligned} f(3) &= 3 \times f(3-1) = 3 \times f(2) = 3 \times 2 = 6 \\ f(2) &= 2 \times f(2-1) = 2 \times f(1) = 2 \times 1 = 2 \\ f(1) &= 1 \times f(1-1) = 1 \times f(0) = 1 \times 1 = 1 \end{aligned}$$

離散数学12

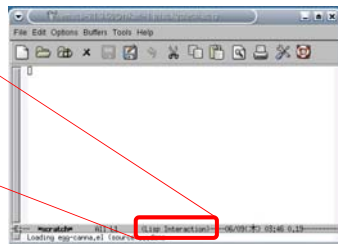
4

12.2 再帰的プログラム (recursive program) の例

関数型プログラミング言語 Lisp による実例

Emacs Lisp の使用法

(Lisp Interaction)



「Esc-x lisp-interaction-mode」と入力

離散数学12

5

再帰的プログラムの例 (階乗計算)

Emacs Lisp による階乗関数の定義

$$(I) \quad f(0) = 1$$

$$(II) \quad f(n) = n \times f(n-1)$$

(defun fact (n)

(cond ((= n 0) 1)

(t (* n (fact (- n 1)))))

C-j

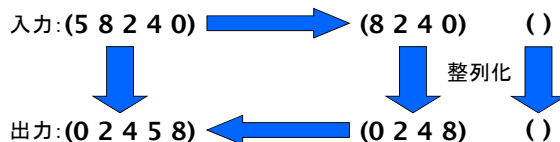
離散数学12

6

再帰的プログラムの例 (整列化)

整列化 (sorting) プログラム

リストの先頭要素を取り除く



取り除いておいた先頭要素(5)を

適切な場所に挿入

挿入法整列化アルゴリズム

離散数学12

7

Lisp による挿入法整列化プログラム

(defun sort (L)

(cond ((null L) ())

(t (insert (car L) (sort (cdr L)))))

(defun insert (x L)

(cond ((null L) (cons x ()))

((<= x (car L)) (cons x L))

(t (cons (car L) (insert x (cdr L)))))

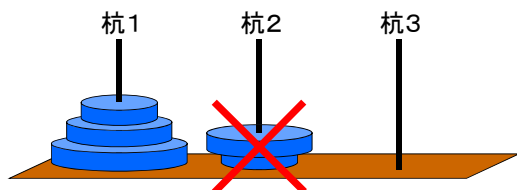
離散数学12

8

12.3 ハノイの塔

n 枚の大きさの異なる円盤を杭1から杭3へ移動せよ

- (1) 1回に1枚だけ動かせる
- (2) 小さい円盤の上に大きな円盤は置けません



離散数学12

10

ハノイの塔を解く再帰的アルゴリズム

$\text{hanoi}(n, \text{from}, \text{to}, \text{work})$: n 枚を from から to へ移動する

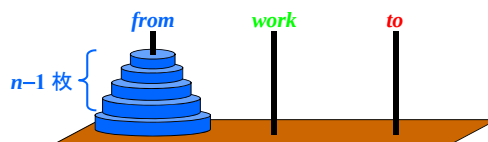
$n = 0 \Rightarrow$ 何もしない

$n \geq 1 \Rightarrow$ 帰納法の仮定: $n-1$ 枚の場合は解ける

$\text{hanoi}(n-1, \text{from}, \text{work}, \text{to})$;

1枚を from から to へ移動 ;

$\text{hanoi}(n-1, \text{work}, \text{to}, \text{from})$;



離散数学12

14

アルゴリズムの効率

$S(n)$: n 枚の円盤を移動させるのに必要なステップ数

$$S(0) = 0$$

$\text{hanoi}(n, \text{from}, \text{to}, \text{work})$:

$\text{hanoi}(n-1, \text{from}, \text{work}, \text{to})$;

1枚を from から to へ移動 ;

$\text{hanoi}(n-1, \text{work}, \text{to}, \text{from})$;

$$S(n) = S(n-1) + 1 + S(n-1) = 2S(n-1) + 1$$

$$S(n) = 2S(n-1) + 1$$

離散数学12

15

13.1 k 階定数係数線形差分方程式

$$f: N \rightarrow X$$

$$C_0 f(n) + C_1 f(n-1) + \dots + C_k f(n-k) = h(n) \quad (*)$$

C_i : 定数, $C_0 \neq 0$, $C_k \neq 0$

例1) $f(n) = f(n-1) + f(n-2)$

$$C_0 = 1, C_1 = -1, C_2 = -1, h(n) = 0$$

2階定数係数線形差分方程式

例2) $f(n) = f(n-1) + n - 1$

$$C_0 = 1, C_1 = -1, h(n) = n - 1$$

1階定数係数線形差分方程式

離散数学12

17

13.2 初期値問題(境界値問題)

$\left\{ \begin{array}{l} k \text{ 階定数係数線形差分方程式} (*) \\ k \text{ 個の連続した } f(n) \text{ の値 (初期値)} \end{array} \right.$

たとえば, $f(0), f(1), \dots, f(k-1)$ の値

→ $(*)$ を満たす関数 $(*)$ の解 を求める

一般解: $(*)$ のすべての解を1つの式で表したもの

$f(n) = 2f(n-1)$ を満たす $f(n)$ は?

$$f(n) = 2^n, f(n) = 2 \cdot 2^n, f(n) = 3 \cdot 2^n, \dots, f(n) = A \cdot 2^n$$

特殊解: $(*)$ のある特定の解

離散数学12

19

13.2 差分方程式の解法

$$C_0 f(n) + C_1 f(n-1) + \dots + C_k f(n-k) = h(n) \quad (*)$$

の右辺 = 0 とおいた式 (同次式)

$$C_0 f(n) + C_1 f(n-1) + \dots + C_k f(n-k) = 0 \quad (**)$$

$(*)$ の一般解 = $(**)$ の一般解 + $(*)$ の1つの特殊解 (同次解)

↓ k 個の初期値

初期値問題の解 (特殊解)

離散数学12

20