2.2 多項式の表現

2.2.1 多項式の表現と加算.

- R : 可換環.
- $R[x, y, \ldots, z]$: 変数 $x, y, \ldots, z$ をもつ R 上の多項式全体.
  ($R[x, y, \ldots, z]$ 上に自然に加減乗算が定義でき, $R[x, y, \ldots, z]$ は可換環となる.)
- 当面は 変数を x : 1変数.
  係数を $\mathbb{Z}$: $\mathbb{Z}[x]$.

Def (主項, 主係数, 次数, モニック)

$$a(x) = a_n x^n + \cdots + a_1 x + a_0 \in \mathbb{Z}[x], \quad a_n \neq 0.$$

$a_n x^n$: $a(x)$ の主項 (the leading term), $lt(a(x))$.
a_n : ______ 主係数 (the leading coefficient), $lc(a(x))$.
n : ______ 次数 (degree), $\deg(a(x))$.
$a(x)$ が モニック (monic) $\overset{\text{def}}{\Longleftrightarrow}$ $lc(a(x)) = 1$. ▨

Def (1変数多項式の表現)

$$a(x) = a_n x^n + \cdots + a_1 x + a_0 \in \mathbb{Z}[x], \quad a_n \neq 0$$

を

$$(a_0, \ldots, a_n) \in \mathbb{Z}^{n+1}$$ で表す. ▨

▨ 1変数多項式の加算

$$a(x) = \sum_{i=0}^{k} a_i x^i = (a_0, a_1, \ldots, a_k),$$

$$b(x) = \sum_{i=0}^{k} b_i x^i = (b_0, b_1, \ldots, b_k)$$

に対し,

$$c(x) = a(x) + b(x) = \sum_{i=0}^{k} c_i x^i = (c_0, c_1, \ldots, c_k)$$

を求める.

9/23

Algorithm (1変数多項式の加算)

入力: $a(x) = \sum_{i=0}^{k} a_i x^i, \quad b(x) = \sum_{i=0}^{k} b_i x^i \in \mathbb{Z}[x]$.

出力: $c(x) = \sum_{i=0}^{k} c_i x^i \in \mathbb{Z}[x]$ s.t. $c(x) = a(x) + b(x)$

(1) for $i \in [0..k]$ do $c_i \leftarrow a_i + b_i$;
(2) return $(c_0, c_1, \dots, c_k)$; ∎

注意

- $\deg(a) \neq \deg(b)$ の場合には、しかるべき対応(拡張)が必要.
- a_i も b_i として、一般に多倍長整数が用いられる.

計算量

$$
\begin{array}{ccccccc}
a(x) & = & (a_0, & a_1, & \dots, & a_k) & \\
+ & & + & + & & + & \leftarrow \text{加算 } k+1 \text{ 回} = \Theta(k). \\
b(x) & = & (b_0, & b_1, & \dots, & b_k) & \\
\downarrow & & \downarrow & \downarrow & & \downarrow & \\
c(x) & = & (c_0, & c_1, & \dots, & c_k) &
\end{array}
$$

$\Theta(\max\{\mathrm{len}(a_0), \mathrm{len}(b_0)\})$　$\Theta(\max\{\mathrm{len}(a_1), \mathrm{len}(b_1)\})$.　$\Theta(\max\{\mathrm{len}(a_k), \mathrm{len}(b_k)\})$

各項の(係数の)多倍長数の長さの和に比例.

- 全体では $\Theta\left(\sum_{i=0}^{k} \max\{\mathrm{len}(a_i), \mathrm{len}(b_i)\}\right)$.
- 係数がすべて単精度ならば $\Theta(k)$.
- 係数がすべて m ワードならば $\Theta(mk)$.

2.2.2 ホーナー (Horner) 法.

◎ 多項式の評価.

$a(x) \in R[x]$, $x_0 \in R$, $a(x) = a_n x^n + \cdots + a_1 x + a_0$ のとき. $a(x_0)$ の値を求める $\longrightarrow$ $x = x_0$ で $a(x)$ の値を評価する.

◎ 素朴な方法.

例 $a(x) = 5x^4 + 3x^3 - 2x^2 + 8x - 10$, $x_0 = 10$ のとき.

$a(10) = 5 \cdot 10^4 + 3 \cdot 10^3 - 2 \cdot 10^2 + 8 \cdot 10 - 10$

乗算回数 4 3 2 1

$\deg(a) = n$ のとき, $\displaystyle\sum_{j=1}^{n} j = \frac{n(n+1)}{2} = O(n^2)$ ⋯ (*)

◎ 剰余定理, 組立除法 $\Rightarrow$ Horner 法.

剰余定理 $a(x) \in R[x]$, $x_0 \in R$

$a(x)$ を $x - x_0$ で割った剰余は $a(x_0)$ に等しい. ▨

組立除法

x_0: 10	5	3	−2	8	−10
		50	530	5280	52880
	5	53	528	5288	$\boxed{52870}$

1倍, 10倍, 10倍, 10倍, 10倍

$= a(10)$.

$a(10) = (((5 \cdot 10 + 3) \cdot 10 - 2) \cdot 10 + 8) \cdot 10 - 10$ と求めている.

より一般に, $a(x) = a_n x^n + \cdots + a_1 x + a_0$ に対し, $a(x_0)$ を求める場合は

$a(x_0) = ((\cdots((a_n \cdot x_0 + a_{n-1}) x_0 + a_{n-2}) x_0 + \cdots) \cdot x_0 + a_1) x_0 + a_0$

と計算する $\rightarrow$ $\boxed{\text{Horner 法 (ホーナー)}}$ = $\boxed{\text{組立除法}}$

(Horner 法は 19世紀初頭. しかし東洋ではより古くから知られていた)

Horner 法のアルゴリズムと計算量.

Algorithm (Horner's rule)

Inputs: $a(x) = a_n x^n + \cdots + a_1 x + a_0 \in \mathbb{Z}[x]$, $x_0 \in \mathbb{Z}$.
Output: $y = a(x_0) \in \mathbb{Z}$.

(1) $y \leftarrow a_n$;
(2) for $i \in [n-1..0]$ do $y \leftarrow y \cdot x_0 + a_i$;
(3) return y;

全体の演算回数は $\Theta(n)$ … 前頁の(☆)に比べて大幅に小さくなっている.
(現時点で, 多倍長整数の乗算の計算量を評価していないことに注意.)

5/30

2.2.3 非負整数の2進・10進変換.

☆ Horner 法のアイデアで効率化できる.

2進表記 ⟶ 10進表記 (整数)

例 $m = (1011011)_2$

$2^6 + 2^4 + 2^3 + 2^1 + 2^0 = (91)_{10}$.

⟺ $M(x) = 1\cdot x^6 + 0\cdot x^5 + 1\cdot x^4 + 1\cdot x^3 + 0\cdot x^2 + 1\cdot x + 1$
に対し, $x \leftarrow 2$ により $M(2)$ を求める.
そこで, $M(2)$ の計算に Horner 法を用いると…

$$
\begin{aligned}
M(2) &= (((((\overset{a_6}{1}\cdot 2 + \overset{a_5}{0})2 + \overset{a_4}{1})2 + \overset{a_3}{1})2 + \overset{a_2}{0})2 + \overset{a_1}{1})2 + \overset{a_0}{1} \\
&= ((((2\cdot 2 + 1)2 + 1)2 + 0)2 + 1)2 + 1 \\
&= (((5\cdot 2 + 1)2 + 0)2 + 1)2 + 1 \\
&= ((11\cdot 2 + 0)2 + 1)2 + 1 \\
&= (22\cdot 2 + 1)2 + 1 \\
&= 45\cdot 2 + 1 \\
&= 91.
\end{aligned}
$$