

5 応用: 多項式の実装

リスト構造を利用して多項式を実装してみよう。簡単の為、整数係数の一変数多項式で変数名は X で固定し、多項式の定数倍や 2 つの多項式の和差積及び整除 (商と余り) を計算することを目標としよう。

とにかく課題として提示しておく。

課題 5 (≠切 学期末迄 (詳しい期日は追って連絡する)). 多項式の型を定義して、入力 (初期値設定)・表示・定数倍・加算・減算・乗算・整除 (商と余りとを求める割り算) を行なう関数を実装し、正しく動作することが一通り確認できるように `main()` 関数でその計算を行なえ。整除については、法 (割る式) の最高次係数が係数環の単数の場合に限って良い。

最初に考えることは、「多項式というデータをどう保持するか」という「データ型の設計」である。なるべく制限を付けたくないで、配列でなくてリストで実装しよう。方針としては次の 2 通りが考えられる。他にもあるかも知れない。

- 係数のリストとして扱う。抜けている次数は係数 0 とする。
- (係数, 次数) の組のリストとして扱う。

例 . $f(X) = 2 - X + 4X^3 + X^5$ の場合、それぞれ、

- 2 -1 0 4 0 1 NULL
- (2,0) (-1,1) (4,3) (1,5) NULL

下の方が通常の表記に近いが、ここでは取り敢えず簡明に上の方で書き始めてみよう。今までの例題では、入力時に 0 を終了の目印としていたが、今度は正当なデータとして 0 が入力されることがあるので、予め次数を入力して指定することにしよう。先頭にダミーノードがあるので、次数をそこに保管すると便利だろう。

提出状況を鑑み、10/28 出題の課題は解答例掲載を延期し、次回掲載の予定。既に提出した人も改良できたら再提出歓迎。

Extra feature 大歓迎。

これを決めないとプログラムが書けないが、書いてみないとどういう設計が適切か (効率が良いとか書き易いとか) が判らない。取り敢えず書いてみるが、後で根本的に書換えることもある、という位のつもりが良いだろう。

降冪・昇冪の選択もあるが、暫定的に昇冪にしてみた。これも実際には、どちらが良いか、書いてみないと判らないかもしれない。

多項式の表示は、体裁をこだわらなければ特に難しくはない。定数倍は係数を全て定数倍するだけ。ベクトルの時と違って次数の異なる多項式同士の加減算もする点や、次数が等しい時に最高次係数が打消しあって次数が下がることがある点には要注意だが、少々気を付ければ加減算までは良いだろう。

実習 5.1. 多項式の型定義・入力 (初期値設定)・表示・定数倍・加算・減算までを実装せよ。加算は $h=f+g$; に相当するものと $f+=g$; に相当するものがある。両方あれば便利だろう。減算・定数倍も同様。

考察 5.1.1. 多項式の乗算および整除 (商と余りを求める計算) の実装の構想を、来週までに練っておこう。整除は最高次係数が単数でないとうまくいかない (余りの次数が下がらない)。有理数型を作ったことがあれば有理数係数にすることも出来ようが、取り敢えず整除に関しては割る式 (法) の最高次係数が 1 と限定して良いことにしておこう。

考察 5.1.2. (係数, 次数) の組のリストとして扱う方針だと、どうすればよいか。また、更に他の方法で何か良さそうなものはあるか。

体裁をこだわらなければ: X^0 とか X^1 とか $0*X^2$ とか $1*X^5$ とかでも良ければ

体上の一変数多項式環は Euclid 整域であるが、 \mathbb{Z} 上の一変数多項式環 $\mathbb{Z}[X]$ は Euclid 整域ではない。

以下、今回は実習時間とするので、今迄の課題の未提出分や実習課題・演習課題に取り組もう。