Regular Paper

可換環の微分の形式化とその応用
Derivation of Commutative Ring and its Application

渡瀬　泰成 [1,*]
Yasushige Watase[1,*]

1 東京都杉並区
1 Suginami-ku, Tokyo, Japan

* yasushige.watase@gmail.com

Proof checked by Mizar Version: 8.1.10 and MML Version: 5.63.1382
Received: December 9, 2020. Accepted: April 30, 2021.

Abstract

A derivation of Polynomial over $\mathbb{R}$ has formalized in [1] and a polynomial was regarded a real differentiable function. In this article we treat more algebraic aspect about derivation rather than analytic case. A derivation is defined for a commutative ring A as a map $D : A \longrightarrow A$ satisfying $D(x+y) = Dx + DyD(xy) = xDy + yDx\ \forall x, y \in A$ [2]. Form the definition various properties of a derivation are derived axiomatic way. In the article most formalized theorems are on to polynomial ring. Because generic theory is rather simpler than polynomial ring case, it is worth to investigate polynomial case for further formalization. There is a certain difficulty in handling with polynomial rings due to an element of the rings has 2 deferent attributes namely 'Polynomial' and 'Element of the carrier of the ring'. The article formalized a finite sequence of polynomial and properties of a derivation of a polynomial ring. We also formalized Leibnitz Formula for power of derivation of $\mathbb{R}[X]$ as example of justification of the current work :

$$D^n(xy) = \sum_{i=0}^{n} \binom{n}{i} D^{n-i} x D^i y.$$

1　はじめに

本稿では, 環の微分を考慮して一変数多項式環 (以下多項式環と記す) の微分とその性質を述べ, 応用として多項式の積の高階微分のライプニッツの公式の形式化を論じた. まず環の微分を定義する.

定義 1 A を可換環とし, 写像 $D : A \longrightarrow A$ が以下をみたすとき D を A の微分という.

$$D(x+y) = Dx + Dy, \quad D(xy) = xDy + yDx \ \forall x, y \in A.$$

と記述される. 恒等零写像は微分の条件を満たす. 恒等零でない具体的な環の微分としては多項式環の微分が挙げられる. これには先行して形式化 [1] がライブラリにある. この形式化は実係数の多項式 p に対して以下の様に形式化されている. しかしながら積の微分は形式化されていない.

Listing 1. POLYDIFFA - Def.5

```
definition
  let p;
  func poly_diff(p) -> sequence of F_Real means
:: POLYDIFF:def 5
  for n being Nat holds it.n = p.(n+1) * (n+1);
end;
```

これに対して可換環上の多項式環の微分で微分の次数をパラメータとして持つ定義をあたえた. 実多項式の形式微分を任意のこの定義では微分は多項式環から多項式環への写像は定義できない. 属性 Polynomial を多項式環の元として扱うことができないことによる. 写像の構成には多項式環の元は属性 Element of the carrier of Polynom-Ring としなければならないが本稿は既存ライブラリと互換性を確保するため POLYDIFF-Def.5 と同様の定義とした.

Listing 2. POLYDIFFA- Def.3

```
definition
  let R;
  let f be Polynomial of R;
  let r be Nat;
  func D_(f,r) -> sequence of R means
POLYDIFFA:def 3
  for i being Nat holds it.i = In((i+r)!/(i!),R) * f.(i+r) ;
end;
```

この定義は既存アーティクル [1] の定義と互換性があり, 多項式の積の微分がライプニッツ則を満たすことが証明される.

Listing 3. POLYDIFFA - Th10&Th31

```
theorem :: POLYDIFFA:10
   for f be Polynomial of F_Real holds D_(f,1) = poly_diff(f);

D1 f ()は D_(f,1)の再定義である.

theorem :: POLYDIFFA:31
     D1(f*'g) = (D1(f))*'g + f*'(D1(g));
     proof
       per cases;
         suppose that
A1: f is non constant;
          thus thesis by A1,Th28;
         end;
         suppose that
A2: f is constant;
          thus thesis by A2,Th27;
         end;
       end;ここで
```

Th27,は以下の如し：Th28

theorem *:: POLYDIFFA:29*
theorem Th27:
for p, q **be** Polynomial **of** F_Real
st p **is** constant **holds** D1(p∗'q) = (D1(p))∗'q + p∗'(D1(q))

theorem *:: POLYDIFFA:30*
theorem Th28:
for x,y **be** Polynomial **of** F_Real **st** x **is non** constant
holds D1(x∗'y) = (D1(x))∗'y + x∗'(D1(y))

2 多項式環の微分の性質の形式化

2.1 基本的な性質の形式化

任意の可換環で微分 D の満たす性質は FM 誌に "RINGDER1" にて形式化した.

i) $D(1_.R) = 0_.R \& D(0_.R) = 0_.R$

ii) $D(nx) = nD(x)$

iii) $Dx^{(n+1)} = (n+1)x^m Dx$

iv) $D^{n+1}x = D(D^n x)$

以下が上記 i) -iv) の多項式に対応した形式化となる.

Listing 4. POLYDIFFA - Th12-15

theorem *:: POLYDIFFA:12*
D1(0._F_Real) = 0._F_Real;

theorem *:: POLYDIFFA:13*
for n,m **be** Nat, f **be** Polynomial **of** F_Real **holds**
In(n,F_Real)∗f + In(m,F_Real)∗f = In(n+m,F_Real)∗f;

theorem *:: POLYDIFFA:14*
D_(f,r) **is** Polynomial **of** F_Real;

theorem *:: POLYDIFFA:15*
D1(D_(f,r)) = D_(f,r+1);

2.2 多項式の微分の公式例

次数 d の多項式 f の 0 次から d 次微分の和は次を満たす.

$$D(f^{(0)} + f^{(1)} + f^{(2)} + \cdots + f^{(d)}) + f = (f^{(0)} + f^{(1)} + f^{(2)} + \cdots + f^{(d)}) \tag{1}$$

この証明概要は任意の多項式の有限列 F に対してその各要素の微分をとる操作を定義し得られる結果を $D(F)$ と表記する. 長さ $d+1$ の多項式の有限列 $\langle f^{(0)}, f^{(1)}, \cdots, f^{(d)}\rangle$ を HWZ(f) として定義する. $d+1$ 番目の成分は微分すると $D(f^{(d+1)}) =$ 恒等 0 写像 となる. 有限列 D(HWZ(f)) の成分と HWZ(f) を比較し和を取り証明される. 形式化では多項式の

有限列は定義出来ないので, Polynomial 属性から Element of the carrier of Polynom-Ring の元への変換関数を定義し多項式 f の多項式環の元のとして表示するとき $\hat{}f$ と表現する. HWZ(f) は $\langle\ \hat{}f^{(0)}, \cdots,\ \hat{}f^{(d)}\rangle$ として扱う．Polynomial 属性から Element of the carrier of Polynom-Ring の元への変換と逆変換の関数を導入する.

定義 2 可換環 A の多項式環を Polynom-Ring A とする．Polynom-Ring A の多項式 f の Element of the carrier of Polynom-Ring A の元への属性の対応を $\hat{}f$ と表記する．Element of the carrier of Polynom-Ring A の元 x を多項式の属性に変換したときは, 変換後の表記を $\tilde{}x$ とする.

上記の定義を踏まえて HWZ(f) は以下の様に定義する.

Listing 5. POLYDIFFA - Def.5

```
definition
  let f be Polynomial of F_Real;
  func HWZ(f) -> FinSequence of the carrier of Polynom-Ring F_Real means
:: POLYDIFFA:def 5

  len it = len f & for i be Nat st i in dom it holds
    it.i = ^(D_(f,i -' 1));
end;
```

the carrier of Polynom-Ring A の元の有限列の総和は既存のライブラリから導入される. しかしながら多項式の有限集合への微分は新規に定義する必要がある.

Listing 6. POLYDIFFA - Def.7

```
definition
  let S be FinSequence of the carrier of Polynom-Ring F_Real;
  func D1(S) -> FinSequence of Polynom-Ring F_Real means
:: POLYDIFFA:def 7

  len it = len S & for i be Nat st i in dom it holds
  it.i = ^D1(~(S/.i));
end;
```

以上の準備の下, 等式 (1) の形式化は以下の如し,

Listing 7. POLYDIFFA - Th39

```
theorem :: POLYDIFFA:39
    for f be non constant Polynomial of F_Real holds
    ~Sum(HWZ(f)) = D1(~Sum(HWZ(f))) + f;
```

2.3　高次冪のライプニッツの公式の形式化

高次冪のライプニッツの公式はよく知られた次の公式である.

$$D^n(fg) = \sum_{i=0}^{n} \binom{n}{i} D^{n-i} f D^i g.$$

この公式をどのように形式的に表現するのか具体的にパスカルの図を作り $D^1(fg), D^2(fg)\cdots$ から公式の形式的な構成を考察する．公式の右辺の項 $\binom{n}{i}D^{n-i}fD^ig$ を成分とする長さ $n+1$ の有限列を考え, それを LBZ と名付け以下の通りに形式化する. FinSequence の引数は集合の元を要請するので FinSequence of Polynomial とは出来ない.

Listing 8. POLYDIFFA - Def.8

```
definition
  let n;
  let x,y be Element of the carrier of Polynom-Ring F_Real;
  func LBZ(n,x,y) -> FinSequence of the carrier of Polynom-Ring F_Real means
:: POLYDIFFA:def 8
  len it = n+1 & for i st i in dom it holds
  it.i = ^(In(n choose (i-'1),F_Real)*((D_(~x,n+1-'i))*'(D_(~y,i -' 1))));
end;
```

この有限列 $D^n(fg) = Sum(\mathrm{LBZ}(D,n,x,y)$ 帰納法にて証明し以下の定理を得る．以下の定理は FM 誌 "RINGDER1" にて形式化した微分の定理の実数上 1 変数の多項式環に対応する形式化であり今後形式化を予定.

Listing 9. RINGDER1 - Th25

```
  theorem Th25:
    (D|^n).(x*y) = Sum(LBZ(D,n,x,y))
theorem :: RINGDER1:25
    (D|^n).(x*y) = Sum(LBZ(D,n,x,y));
```

証明は $\mathrm{LBZ}(D,n,x,y)$ は $\binom{n}{i}D^{n-i}fD^ig$ に対応している．但し形式化では添え字のずれを補正している．ここでは非形式での証明及び図 1 のパスカルの図との対応を示し形式表現の読解の補いとしたい． LBZ1 と LBZ2 は組み合わせの数の分解の部分 $\binom{n}{i}D^{n-i}fD^ig = (\binom{n}{i} + \binom{n}{i+1})D^{n-i}fD^{i+1}g$ と変形したときの二つの項の和を二つの項にして有限列にし，この組み合わせの数の分解の前半, 後半の項で生成できる列の先頭と末尾の項を除いて奇数項、偶数項をとってそれぞれ LBZ1 と LBZ2 部分列に分けている．これらを準備して帰納法で $\mathrm{LBZ}(D,n,x,y) = \sum_{i=0}^{n}\binom{n}{i}D^{n-i}fD^ig$ が n の時成り立てば $n+1$ でも成り立つことを証明する．計算手順は Σ 記号の添え字と項の式変形の計算で形式表現可能と考えられるがここではパスカルの図の手順にならっている．

Listing 10. POLYDIFFA - Def.9-11

```
definition
  let m;
  let x,y be Element of Polynom-Ring F_Real;
  func LBZ0(m,x,y) -> FinSequence of the carrier of Polynom-Ring F_Real means
:: POLYDIFFA:def 9
  len it = m & for i st i in dom it holds
  it.i =^(In((m choose (i-'1))+(m choose i),F_Real)*
        ((D_(~x,m+1 -'i))*'(D_(~y,i))));
end;

definition
  let m;
  let x,y be Element of the carrier of Polynom-Ring F_Real;
  func LBZ1(m,x,y) -> FinSequence of the carrier of Polynom-Ring F_Real means
:: POLYDIFFA:def 10
  len it = m &
  for i st i in dom it holds
  it.i = ^(In(m choose (i-'1),F_Real)*((D_(~x,m+1 -'i))*'(D_(~y,i))));
end;

definition
  let m;
  let x,y be Element of the carrier of Polynom-Ring F_Real;
  func LBZ2(m,x,y) -> FinSequence of the carrier of Polynom-Ring F_Real means
:: POLYDIFFA:def 11
  len it = m &
  for i st i in dom it holds
```

```
    it.i = ^(In(m choose i,F_Real)*((D_(~x,m +1 -'i))*'(D_(~y,i))));
end;
```

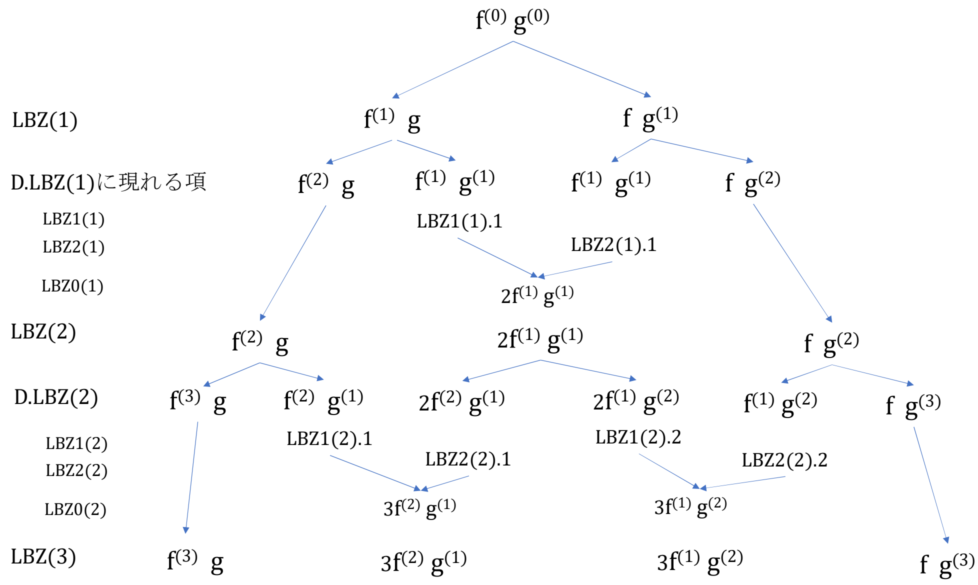

図 1. 多項式 f,g の微分のパスカルの図. $f^{(i)}, g^{(j)}$ はそれぞれ f, g の i 次,j 次の微分を表す. LBZ はライプニッツ公式に現れる項を与える多項式の有限列.

3　形式化の課題と展望

多項式の微分に関する定理を形式化する場合, 証明の記述の途中で属性の変換が発生, 証明自体分かりにくいものとなる. 多項式に関するライブラリの既存資源は多項式を Polynomial 属性で扱って居り多項式環の元から出発した定義との親和性が低い点が課題といえる. 2.2 節の定義 2 の属性変換を利用せざる得なかった. 今回は $\mathbb{R}[X]$ の多項式を主に扱ったが一般の可換環係数の多項式環でも同様に形式化できる. 但し多項式に整数を乗ずる場合で係数環の有限標数では扱い異なる. 今回の形式化により多項式とその微分を伴った定理群の証明が可能となるのでその応用は広い. 更なる発展としては有理関数体の微分やべき級数, ローラン級数の微分などが見込まれる.

参考文献

[1] Korniłowicz A. Differentiability of Polynomials over Reals. Formalized Mathematics. 2017;25(**1**):31–37.

[2] Matsumura H. Commutative Ring Theory. 2nd ed. Cambridge Studies in Advanced Mathematics. Cambridge University Press; 1989.

Mizar article information

Works in Progress

POLYDIFFA Properties of Modules over a Commutative Ring
by Yasushige Watase

Summary: A derivation of Polynomial over $\mathbb{R}$ has formalized in [1] and a polynomial was regarded a real differentiable function. In this article we treat more algebraic aspect about derivation rather than analytic case. A derivation is defined for a commutative ring A as a map $D : A \longrightarrow A$ satisfying $D(x+y) = Dx+DyD(xy) = xDy+yDx \; \forall x, y \in A$ [2]. Form the definition various properties of a derivation are derived axiomatic way. In the article most formalized theorems are on to polynomial ring. Because generic theory is rather simpler than polynomial ring case, it is worth to investigate polynomial case for further formalization. There is a certain difficulty in handling with polynomial rings due to an element of the rings has 2 deferent attributes namely 'Polynomial' and 'Element of the carrier of the ring'. The article formalized a finite sequence of polynomial and properties of a derivation of a polynomial ring. We also formalized Leibnitz Formula for power of derivation of $\mathbb{R}[X]$ as example of justification of the current work :

$$D^n(xy) = \sum_{i=0}^{n} \binom{n}{i} D^{n-i}xD^iy.$$

Listing 11. POLYDIFFA - abstract

```
environ

 vocabularies VECTSP_1, POLYNOM1, PARTFUN1, NUMBERS, REAL_1, RELAT_1, TARSKI,
      STRUCT_0, FUNCT_1, SUBSET_1, ALGSEQ_1, NAT_1, XXREAL_0, SUPINF_2, INT_3,
      POLYNOM3, ARYTM_3, CARD_1, FUNCT_7, MESFUNC1, FINSEQ_1, INT_1, XBOOLE_0,
      HURWITZ, ARYTM_1, AFINSQ_1, REALSET1, CARD_3, ORDINAL4, XCMPLX_0, C0SP1,
      POLYDIFF, FINSEQ_3, FIELD_1, RFINSEQ, RATFUNC1, XREAL_0, RAT_1, GAUSSINT,
      POLYNOM5, VECTSP_2, FOMODEL1, ALGSTR_0, ZFMISC_1, FINSEQ_2, FINSEQ_5,
      POLYDIFA;

 notations TARSKI, XBOOLE_0, ZFMISC_1,SUBSET_1, RELAT_1, FUNCT_1, ORDINAL1,
      RELSET_1, PARTFUN1, FUNCT_2, BINOP_1, FUNCT_3, FUNCOP_1, NUMBERS,
      XCMPLX_0, XXREAL_0, XREAL_0, NAT_1, INT_1, RAT_1, VALUED_0, INT_2, NAT_D,
      FINSEQ_1, FINSEQ_2, NEWTON, RFINSEQ, FINSEQ_5, STRUCT_0, ALGSTR_0,
      RLVECT_1, VECTSP_1, COMPLFLD, GROUP_1, FUNCSDOM, VECTSP_2, GROUP_4,
      NORMSP_1, VFUNCT_1, FVSUM_1, POLYNOM1, ALGSEQ_1, POLYNOM3, POLYNOM5,
      UPROOTS, INT_3, HURWITZ, C0SP1, EC_PF_1, RATFUNC1, GAUSSINT, RING_3,
      RING_4, POLYDIFF, FIELD_1, FIELD_4;

 constructors TARSKI, XBOOLE_0, ZFMISC_1, SUBSET_1, RELAT_1, FUNCT_1, ORDINAL1,
      RELSET_1, PARTFUN1, FUNCT_2, BINOP_1, FUNCT_3, REALSET1, XCMPLX_0,
      FINSET_1, NUMBERS, SQUARE_1, NAT_1, INT_1, RAT_1, INT_2, NAT_D, BINOP_2,
      NEWTON, RFINSEQ, FINSEQ_5, FUNCT_7, STRUCT_0, ALGSTR_0, RLVECT_1,
      VECTSP_1, GROUP_1, COMPLFLD,
      VECTSP_2, VFUNCT_1, GROUP_4, FVSUM_1, POLYNOM1, ALGSEQ_1, BINOM,
      POLYNOM3, POLYNOM4, POLYNOM5, UPROOTS, INT_3, C0SP1, VECTSP11, EC_PF_1,
      RATFUNC1, GAUSSINT, RING_3, POLYDIFF, FIELD_1, FIELD_4;
```

```
registrations RELSET_1, FUNCT_2, VECTSP_1, STRUCT_0, XBOOLE_0, RELAT_1,
    FUNCT_1, NAT_1, XREAL_0, NUMBERS, XCMPLX_0, ORDINAL1, POLYNOM3, MEMBERED,
    FUNCOP_1, RATFUNC1, INT_1, POLYNOM5, FINSEQ_1, VALUED_1, VALUED_0,
    FDIFF_1, RCOMP_1, POLYNOM4, RLVECT_1, XXREAL_0, ALGSTR_1, POLYDIFF,
    RING_4, AFINSQ_1, NEWTON, GAUSSINT, COMPLFLD, RAT_1, INT_3, FIELD_4,
    RING_5, FINSEQ_5;

requirements SUBSET, BOOLE, NUMERALS, REAL, ARITHM;

definitions TARSKI, FUNCT_1, FUNCT_2, FDIFF_1, ALGSEQ_1, RATFUNC1, XBOOLE_0,
   RELAT_1, RELAT_2, RLVECT_1, GROUP_1, VECTSP_1, ALGSTR_0;

equalities SUBSET_1, STRUCT_0, ALGSTR_0, VECTSP_1, POLYNOM3, HURWITZ,
   POLYNOM5, FUNCSDOM, VALUED_1, FINSEQ_1, XCMPLX_0, POLYDIFF, RLVECT_1,
   RATFUNC1, BINOP_1, FINSEQ_2, XBOOLE_0, NUMBERS, XREAL_0, INT_3, COMPLEX1,
   RING_4, FUNCOP_1, SQUARE_1, PARTFUN1;

expansions FDIFF_1, ALGSEQ_1, FUNCT_1, POLYNOM1, NUMBERS,
   STRUCT_0, VECTSP_1, TARSKI, FUNCT_2, FINSEQ_1, RING_4, RFINSEQ,
   RATFUNC1, POLYDIFF, RLVECT_1, GROUP_1, SUBSET_1,XREAL_0, NORMSP_1,
   SQUARE_1, XBOOLE_0, VECTSP_2, IDEAL_1;

theorems FUNCT_2, POLYNOM5, UPROOTS, FINSEQ_5, ORDINAL1, XREAL_0, ALGSEQ_1,
   XXREAL_0, XREAL_1, NAT_1, RATFUNC1, HURWITZ, INT_1, FINSEQ_1, FINSEQ_3,
   RLVECT_1, NORMSP_1, PARTFUN1, NAT_D, FVSUM_1, POLYNOM4, NEWTON, POLYNOM3,
   POLYDIFF, XCMPLX_1, BINOM, RING_4, RFINSEQ, FIELD_1, RING_5, FUNCT_1,
   SUBSET_1, FINSEQ_2, FINSEQ_4, PRE_POLY, TARSKI, INT_3, FIELD_4, C0SP1,
   POLYALG1, VECTSP_2, POLYNOM1, FUNCT_3, ZFMISC_1;

schemes FUNCT_2, NAT_1, FINSEQ_1;

begin :: Preliminaries
reserve R1 for non degenerated comRing;
reserve R for (INT.Ring)-extending comRing;

reserve c for Complex;
reserve x0,r for Real;
reserve i,j,k, m,n,d,r for Nat;
reserve x,y,z for Element of the carrier of Polynom-Ring F_Real;

registration
  cluster F_Real -> (INT.Ring)-extending;
end;

registration
  let R;
  let i be Integer;
  reduce In(i,R) to i;
end;

reserve f for Element of the carrier of Polynom-Ring R;

theorem :: POLYDIFFA:1
  for p be Polynomial of R holds p`^n is Polynomial of R;

definition
  let R1;
  let p be Element of the carrier of Polynom-Ring R1;
  func ~p -> Polynomial of R1 means
:: POLYDIFFA:def 1

  it is Polynomial of R1 & it = p;
end;

definition
  let R1;
  let p be Polynomial of R1;
  func ^p -> Element of the carrier of Polynom-Ring R1
 means
```

```
:: POLYDIFFA:def 2

  it is Element of the carrier of Polynom-Ring R1 & it = p;
end;

theorem :: POLYDIFFA:2
   for p,q be Polynomial of R1 holds ^(p+q) = (^p) + (^q);

theorem :: POLYDIFFA:3
   for p be Element of the carrier of Polynom-Ring R1 holds ^(~p) = p;

theorem :: POLYDIFFA:4
   for p be Polynomial of R1 holds ~(^p) = p;

theorem :: POLYDIFFA:5
   for p,q be Element of Polynom-Ring R holds
   ~(p+q) = (~p) + (~q);

theorem :: POLYDIFFA:6
   for p,q be Element of the carrier of Polynom-Ring R holds
   (p+q) = (~p) + (~q);

theorem :: POLYDIFFA:7
   for p,q be Element of Polynom-Ring R holds
   ~(p*q) = (~p) *' (~q);

theorem :: POLYDIFFA:8
   for p,q be Polynomial of R1 holds ^(p*'q) = (^p) * (^q);

theorem :: POLYDIFFA:9
   for p,q be Polynomial of F_Real holds ^(p*'q) = (^p) * (^q);

definition
  let R;
  let f be Polynomial of R;
  let r be Nat;
  func D_(f,r) -> sequence of R means
:: POLYDIFFA:def 3

  for i being Nat holds it.i = In((i+r)!/(i!),R) * f.(i+r) ;
end;

registration
  let R;
  let f be Polynomial of R;
  let r be Nat;
  cluster D_(f,r) -> finite-Support;
end;

theorem :: POLYDIFFA:10
   for f be Polynomial of F_Real holds
   D_(f,1) = poly_diff(f);

definition
  let f be Polynomial of F_Real;
  func D1(f) -> Polynomial of F_Real equals
:: POLYDIFFA:def 4

  poly_diff(f);
end;

reserve f for Polynomial of F_Real;

theorem :: POLYDIFFA:11
   D_(f,0) = f;

theorem :: POLYDIFFA:12
    D1(0_F_Real) = 0_F_Real;
```

```
theorem :: POLYDIFFA:13
  for n,m be Nat, f be Polynomial of F_Real holds
  In(n,F_Real)*f + In(m,F_Real)*f = In(n+m,F_Real)*f;

theorem :: POLYDIFFA:14
  D_(f,r) is Polynomial of F_Real;

theorem :: POLYDIFFA:15
  D1(D_(f,r)) = D_(f,r+1);

theorem :: POLYDIFFA:16
   for f be non constant Polynomial of F_Real holds
   deg f >= 1 & len f >= 2 & len f = deg f + 1 & len f-'1 = len f -1 &
   len f > 1;

reserve p for Polynomial of F_Real;
reserve p for Polynomial of F_Real;
reserve f for Polynomial of F_Real;

theorem :: POLYDIFFA:17
   n = len f implies len (D_(f,n)) = 0;

theorem :: POLYDIFFA:18
   for f be Polynomial of F_Real holds D_(f,len f) = 0_.F_Real;

theorem :: POLYDIFFA:19
   n = len f -1 implies len (D_(f,n)) = 1;

theorem :: POLYDIFFA:20
   n = len f -1 implies D_(f,n) = <% In(n!,F_Real)*f.n %>;

theorem :: POLYDIFFA:21
 for p being constant Polynomial of F_Real holds D_(p,1) = 0_.F_Real;

theorem :: POLYDIFFA:22
 for p being constant Polynomial of F_Real holds D_(p,1) = 0_.F_Real;

reserve a for Element of F_Real;

theorem :: POLYDIFFA:23
    for i be Nat, p be Polynomial of F_Real
    holds ((a|F_Real)*'p).i = a*p.i;

theorem :: POLYDIFFA:24
   for f,g being Polynomial of F_Real for a be Element of F_Real
   st f = a|F_Real holds D1(f*'g) = (a|F_Real)*'(D1(g));

theorem :: POLYDIFFA:25
   for f being Polynomial of F_Real for a be Element of F_Real
   st f = anpoly(a,0) holds D1(f) = 0_.F_Real;

theorem :: POLYDIFFA:26
   for f being Polynomial of F_Real for a be Element of F_Real
   st f = anpoly(a,1) holds D1(f) = anpoly(a,0);

reserve f,g for Polynomial of F_Real;

theorem :: POLYDIFFA:27
    for f,g st f = anpoly(1.F_Real,1) holds
    for i be Element of NAT holds (f*'g).(i+1) = g.i &(f*'g).0 = 0.F_Real;

theorem :: POLYDIFFA:28
    for f,g st f = anpoly(1.F_Real,1) holds
    D1(f*'g) = (D1(f))*'g + f*'(D1(g));

theorem :: POLYDIFFA:29
   for p, q be Polynomial of F_Real
   st p is constant holds D1(p*'q) = (D1(p))*'q + p*'(D1(q));
```

```
theorem :: POLYDIFFA:30
   for x,y be Polynomial of F_Real st x is non constant
   holds D1(x*'y) = (D1(x))*'y + x*'(D1(y));

theorem :: POLYDIFFA:31
   D1(f*'g) = (D1(f))*'g + f*'(D1(g));

theorem :: POLYDIFFA:32
   for n,f holds D1(In(n,F_Real)*f) = In(n,F_Real)*D1(f);

theorem :: POLYDIFFA:33
    for f be Polynomial of F_Real holds
    D1(f`^(m+1)) = In(m+1,F_Real)*(f`^m)*'D1(f);

reserve f for non constant Polynomial of F_Real;

reserve a for Element of F_Real;

::::::::::::::::::::::::::::::::::::::::::::::::::::
::: F + F'+F"+ ....F^(n) - D(F + F'+F"+ ....F^(n)) = F
::::::::::::::::::::::::::::::::::::::::::::::::::::
::reserve f for non constant Element of the carrier of Polynom-Ring F_Real;
::L-polynomial-membered

definition
  let f be Polynomial of F_Real;
  func HWZ(f) -> FinSequence of the carrier of Polynom-Ring F_Real means
:: POLYDIFFA:def 5

  len it = len f & for i be Nat st i in dom it holds
    it.i = ^(D_(f,i -' 1));
end;

definition
  let f be Polynomial of F_Real;
  func HWZ2(f) -> FinSequence of the carrier of Polynom-Ring F_Real means
:: POLYDIFFA:def 6

  len it = len f & for i be Nat st i in dom it holds
    it.i = ^(D_(f,i));
end;

definition
  let S be FinSequence of the carrier of Polynom-Ring F_Real;
  func D1(S) -> FinSequence of Polynom-Ring F_Real means
:: POLYDIFFA:def 7

  len it = len S & for i be Nat st i in dom it holds
  it.i = ^D1(~(S/.i));
end;

theorem :: POLYDIFFA:34
   for S be FinSequence of the carrier of Polynom-Ring F_Real,
   v be Element of the carrier of Polynom-Ring F_Real holds
   D1(S^<*v*>) = (D1(S))^<*^D1(~v)*>;

theorem :: POLYDIFFA:35
   for S be FinSequence of the carrier of Polynom-Ring F_Real holds
   D1(~(Sum S)) = ~Sum( D1(S));

theorem :: POLYDIFFA:36
    (HWZ(f)).(len f) = <% In((deg f)!,F_Real)*f.(deg f) %> &
    poly_diff(~((HWZ(f))/.(len f))) = 0_.F_Real;

theorem :: POLYDIFFA:37
   for f be non constant Polynomial of F_Real holds
   HWZ(f) = <*^f*>^Del((HWZ2(f)),len f);

theorem :: POLYDIFFA:38
```

```
    D1(HWZ(f)) = HWZ2(f);

theorem :: POLYDIFFA:39
    for f be non constant Polynomial of F_Real holds
    ~Sum(HWZ(f)) = D1(~Sum(HWZ(f))) + f;

reserve x,y for Element of the carrier of Polynom-Ring F_Real;
definition
    let n;
    let x,y be Element of the carrier of Polynom-Ring F_Real;
    func LBZ(n,x,y) -> FinSequence of the carrier of Polynom-Ring F_Real
    means
:: POLYDIFFA:def 8

    len it = n+1 & for i st i in dom it holds
    it.i = ^(In(n choose (i-'1),F_Real)*((D_(~x,n+1-'i))*'(D_(~y,i -' 1))));
end;

theorem :: POLYDIFFA:40
      LBZ(0,x,y) = <* x*y *>;

theorem :: POLYDIFFA:41
        LBZ(1,x,y) = <* y* (^D1(~x)), x* (^D1(~y)) *>;

definition
    let m;
    let x,y be Element of Polynom-Ring F_Real;
    func LBZ0(m,x,y) -> FinSequence of the carrier of Polynom-Ring F_Real
    means
:: POLYDIFFA:def 9

    len it = m & for i st i in dom it holds
    it.i =^(In((m choose (i-'1))+(m choose i),F_Real)*
            ((D_(~x,m+1 -'i))*'(D_(~y,i))));
end;

definition
    let m;
    let x,y be Element of the carrier of Polynom-Ring F_Real;
    func LBZ1(m,x,y) -> FinSequence of the carrier of Polynom-Ring F_Real
    means
:: POLYDIFFA:def 10

    len it = m &
    for i st i in dom it holds
    it.i = ^(In(m choose (i-'1),F_Real)*((D_(~x,m+1 -'i))*'(D_(~y,i))));
end;

definition
    let m;
    let x,y be Element of the carrier of Polynom-Ring F_Real;
    func LBZ2(m,x,y) -> FinSequence of the carrier of Polynom-Ring F_Real
    means
:: POLYDIFFA:def 11

    len it = m &
    for i st i in dom it holds
    it.i = ^(In(m choose i,F_Real)*((D_(~x,m +1 -'i))*'(D_(~y,i))));
end;

reserve p1,q1 for FinSequence of the carrier of Polynom-Ring F_Real;

theorem :: POLYDIFFA:42
    LBZ0(m,x,y) = LBZ1(m,x,y) + LBZ2(m,x,y);

theorem :: POLYDIFFA:43
    Sum LBZ0(n,x,y) = Sum LBZ1(n,x,y) + Sum LBZ2(n,x,y);

theorem :: POLYDIFFA:44
```

D1(LBZ0(n,x,y)) = Del(LBZ2(n+1,x,y),n+1) + Del(LBZ1(n+1,x,y),1);

theorem *:: POLYDIFFA:45*
Sum(D1(LBZ0(n,x,y))) =
− (LBZ1(n+1,x,y)/.1) + Sum LBZ0(n+1,x,y) − (LBZ2(n+1,x,y)/.(n+1));

theorem *:: POLYDIFFA:46*
LBZ(n+1,x,y) = <*^((D_(~x,n+1))*'~y)*>^(LBZ0(n,x,y))^<*^(~x*'(D_(~y,n+1)))*>
;

theorem *:: POLYDIFFA:47*
Sum(<* ^((D_(~x,n+1))*'~y) *>^(LBZ0(n,x,y))^<* ^(~x*'(D_(~y,n+1))) *>)
= ^((D_(~x,n+1))*'~y) + Sum(LBZ0(n,x,y))+ ^(~x*'(D_(~y,n+1)));