数理解析研究所講究録
第 581 巻 1986 年 131-145

ＩＣＡＳ機能の作成

対　馬　勝　英　，佐　藤　季　弘　（大阪電気通信大学）
Tsushima Katsuhide　Satou Toshihiro

ＩＣＡＳは、紙と鉛筆の上で行う記号法と処理のアルゴリスムをできる限り保存しつつ数式処理を行うという設計理念の元に作られた。また、パーソナルコンピュータを用いて数理研究用のパーソナルなツールを構築するという小型数式処理システムの枠内でそれを実現したものが、ｍｕ－ＭＡＴＨ上のＩＣＡＳ機能である。

さらに、やっと普及し始めた68000 をCPU とする高位パソコン上で上記を利用するために全く新しく　ICAS-68　を作成した。

その意味では、mu-MATH 上のICASは上記の機能を容易に実現しつつ、それを実験的に検証するという実験的システムの色彩を持ち、ICAS68はそれを本格化したシステムである。

我々はICASと呼ばれる数式処理機能を設計し、実現し報告したが、それに関して必ずしも十分な理解が得られずまた幾つかの反論もあったので、始めて設計理念を述べ通常の数式処理システムとの概念的な差を明確にしたい。

数学的な実体の定義

量子力学的散乱理論の研究を行うため、我々はオペレタを明示的に扱うことのできる数式処理システムを必要とした。それを具体化するには、数式処理システムに新しい情報構造を持たすという大巾な変更を必要とした。従来の数式処理システムでは、

$$\frac{d}{dx}x^2 = 2\ x$$　を

DIF(x^2 ,x)-→2* x

と入力して実現する。

関数 $F(x) = x^2$　は

```
FUNCTION F(x)
  x∧2  ,
ENDFUN$  .
```

と定義でき

```
DIF((F(x),x) = 2* x
```

を得る。（このときF(x)は数学的関数であると共にＬＩＳＰ関数でもある。また，ＤＩＦもＬＩＳＰ関数であり，ＤＩＦとF(x)は同様な情報構造として扱われている。）

ＩＣＡＳではオペレタ$\frac{d}{dx}$　は

```
DEFOP Dx  ,DIF(#EXP,x)  ,ENDOP;
```

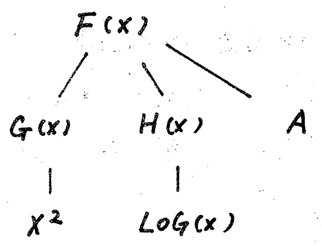
数学的関数　$F(x) = x^2$　は

```
DEFINE  F(x)  ,x  2,ENDDEF;
```

独立変数　$x = yz$　は

```
DEFVAR  x,y*z,ENDVAR;
```

と別個に定義する。この様に情報構造を増やし、独立した実体とすることで後にのべる柔軟な処理能力と表記能力が獲得できる。



<u>階層的な定義と中間的表現</u>

```
F(x) = G(x)+H(x)+A
G(x) = x 2
H(x) = LOG
```

と、関数F(x),G(x),H(x)が上図の様に階層的に定義されているとき、紙の上では我々はそれをどのレベルでも見ることができるし利用できる。

```
FUNCTION F(x)
G(x)+H(x)+A  ,
ENDFUN$                                              ①
```

と定義し、G(x),H(x) を

```
FUNCTION G(x) ,x∧2  , ENDFUN $                       ②
FUNCTION H(x) ,LOG(x) ,ENDFUN $                      ③
```

と定義すると、

```
F(x);  ————>  x∧2+LOG(x)+A                           ④
```

を得るが、

```
F(x);  ————>  F(x)                                   ⑤
       ————>  G(x)+H(x)                              ⑥
```

を得ることはできない。(②，③を定義する前では確かに⑥が得られるが、それではどのレベルの表式も定義の前後に関係無くいつでも取り出して利用できることにならず、紙の上の様な自由な取扱いが保証されたことにはならない。)

④，⑤，⑥が自由に行えることを、我々は関数の『評価の抑止』と呼んでいる。(この機能はICASにおいては階層の数に依存せず利用できる。)

この評価の抑止は微分を行った場合，

```
DIF(F(x),x)  ————>  Fx(x)
             ————>  Gx(x)+Hx(x)
             ————>  2+x+1/x
```

の3 種の出力を許す原因となっている。このことより、評価の抑止は単なる結果の表示に関わるものでなく、任意の内部的処理に関わるものであることが理解される。従って，ICASの評価抑止機能を出力にＬＥＴを用いたうわべだけの処理と混同しないで戴きたい。これについては付録のＫＡＩ，ＰＡＤＥのリストを熟読されたい。

この様に、オペレタ，関数，独立変数の各々に対して評価レベルの抑止が自由に行えることにより、後に示す様な多彩な出力を得る。

オペレタ

さて、ここでオペレタを取り扱えるということの意味を定義したい。

$$\{(\theta_1+\theta_2)^2+(\theta_1-\theta_2)^2\}\ f(x)=(2\theta_1^2+2\theta_2^2)f(x) \quad ①$$

$$\frac{d}{dx}(\int f(x)\,dx)=\frac{d}{dx}\cdot(\int dx)\cdot f(x)=f(x) \quad ②$$

関数の不定積分と微分を連結し作用させると、恒等的オペレタを作用させたことになる。

①はオペレタが整式と同じ簡単化規則により扱えることを利用して、オペレタ代数を行った。これを数式処理システム上で行うには、オペレタが数学的関数，整式とは区別した独立した実体として定義されていて、それに対して整式と同じ簡単化ルールを適用できる様な情報構造を設計する必要がある。

一方、②は$\int dx$ と$\frac{d}{dx}$が逆演算であるという知識が、前置型の形式の内で適用できる必要がある。

一般に人間がオペレタを利用するということは、前置型表記法が可能であることが前提となる。

例えば、

$$\left(\frac{d}{dx}+\frac{d^2}{dx^2}\right)^2 U(x) \quad ③$$

を人間が処理して

$$\left(\frac{d^2}{dx^2}+2\frac{d^3}{dx^3}+\frac{d^4}{dx^4}\right)U\ (x) \quad ④$$

として

```
DIF(U(x),x,2)+2*DIF(U(x),x,3)+DIF(U(x),x,4)
```
⑤

としなければ処理できないREDUCEやmu-MATHHなどは、オペレタを前置型で扱う機能を持つと言えない。

人間に③をそのまま入力させるのでなく、④の前処理をして⑤の入力を行うのでは、紙と鉛筆での記号法と数式処理システム上の記号法が異なると同時に、アルゴリスムも変化してしまう。これでは冒頭に述べた設計理念は達成できない。

我々の作成したICASでは

```
(Dx+Dxx)∧2. U(x)  ;
```
⑥

と入力でき、③の表記法とアルゴリスムは変更されない。

ただし、Dxは $\frac{d}{dx}$，Dxx は$\frac{d^2}{dx^2}$という微分オペレタであり、これらは事前に次の様にオペ

レタ定義機能により対話的に定義する。

```
DEFOP  Dx  ,DIF(#EXP,x)  ,ENDOP;         ⑦
DEFOP  Dxx ,DIF(#EXP,x,2),ENDOP;         ⑧
```

従って、①を実現するには $0_1, 0_2$ が

```
DEFOP  01 , ENDOP;                       ⑨
DEFOP  02 , ENDOP;                       ⑩
```

と空で定義されていて,

```
( (01+02)^2+(01-02)^2).F(x)              ⑪
```

と入力すると、

```
2*01^2.F(x)+2*02^2.F(x)                  ⑫
```

を得る。

ここで⑫ が得られるためには、未定義関数の形式的処理が行えることと、オペレタ代数が行えることが要求される。

未定義な実体 の形式的処理は関数のみならず、オペレタに対しても許される必要があり ⑫はそれを利用してえられた。(0 の具体的な内容が定義されていて，その評価を抑止した場合にも同様な結果が得られる。)

これはLISPの観点よりすると、評価の抑止機能をオペレタ，関数，独立変数に対して実現することに他ならない。そのためにもオペレタ，関数，独立変数が単なるリスト構造でなく、独立した個々の情報構造として設計されている必要がある。

LISP関数としてDIF があることで、通常の数式処理システムはオペレタを扱えると即断していけない。LISP関数であるオペレタを処理するオペレタ処理のためのドライバが用意されていなければ、オペレタは数式処理システム上で「生きて」いるとは言えない。

この意味で，整式の簡単化のLISP関数は持つがオペレタの簡単化のLISP関数は持たないmu-MATH,REDUCE等の数式処理システムは、オペレタを明示的に扱っている訳ではない。

以上の分析をたどることにより、我々はオペレタを明示的に扱う機能を数式処理システムmu-MATH に組み込んだ。

またそのために、オペレタ，数学的関数，独立変数を独立した情報構造として定義した。それらの情報構造の独立性より、任意のレベルでのオペレタ，数学的関数，独立変数の

評価の抑止が可能となった。　　　　単に出力として　　　　種々のものが得られるのにとどまらず、内部処理そのものについての抑止を行えることに注意したい。（これについてはＰＡＤＥのリストを参照されたい。）

任意の常微分方程式のPADE近似

$$\frac{d^2u}{dx^2} = f\left(u, \frac{du}{dx}, x\right)$$

なる微分方程式の巾級数解

$$u(x) = c_0 + c_1x + c_2x^2 + \cdots + c_nx^n$$

を求める　KAI　パッケヂ，それを用いてそのPade展開を求めるパッケヂ　PADE　を用いて、線型，非線型を問わずPade近似を得ることができる。

さらに、

$$\frac{d^nu}{dx^n} = f\left(\frac{d^{n-1}u}{dx^{n-1}}, \cdots, \frac{du}{dx}, u, x\right) \quad や,$$

$$\begin{cases} \dfrac{d^2u}{dx^2} = f\left(u, \omega, \dfrac{du}{dx}, \dfrac{d\omega}{dx}, x\right) \\ \dfrac{d^2\omega}{dx^2} = g\left(u, \omega, \dfrac{du}{dx}, \dfrac{d\omega}{dx}, x\right) \end{cases}$$

等の連立微分方程式に対してもKAI によりそのPADE近似を求めることができる。

このパッケヂ自身の有用性は言うまでもないが、特にPADEの部分はMACSYMA によるそれと意識的に同じ方式を採用して記述した。

これより、ICASの持つ処理機能の高さが理解できる。MACSYMA のソースが公開されぬのでその実現手法については判断できないが、ソースの公開された我々の ICAS　でも同等のことが実行できることを示せた。

ＩＣＡＳ機能（ｍｕ－ＭＡＴＨ版）の特徴

ICAS　の特徴として

① LETが使える。

② FORが使える。

③非可換数が扱える。

④より複雑ルールを対話登録するMAKERULE機能を持つ。

⑤形式積分が行える。

⑥諸種の情報量の現状がユーティリティにより見れる。

があるが、④について以下に述べたい。

MAKERULE

ＩＣＡＳの特徴の一つである対話的なルール登録機能ＭＡＫＥＲＵＬＥについて形式積分機能ＳＩＮＴを例にとり，ＬＥＴと比較して述べてみよう。

ＩＣＡＳは形式微分機能ＳＤＩＦの反対の機能として形式積分機能ＳＩＮＴを持つ。

$$\int u_x(x)\,dx = u(x) \qquad ①$$

$$\iiint u_{xxxx}(x)\,dx\,dx\,dx = u_x(x) \qquad ②$$

の様な不定積分を行うオペレタ～を定義し，それを前置形式で利用したい。

その為にmu-MATH の積分機能を一部変更したＳＩＮＴ関数を作り，それを

```
DEFOP ～ ,SINT(#EXP,x),ENDOP;
```

により前置形式のオペレタ　と定義する。

①は　　～. $u_x(x)$;　----->　$u(x)$

②は　　(～^3). $u_{xxxx}(x)$;　--->　$u_x(x)$

としてＩＣＡＳ上で実現される。（この様な積分機能をＳＩＮＴ機能と呼んでいる。）

しかし，

$$\int \frac{du}{dx}\,u\,dx = \frac{1}{2}u^2 \qquad ③$$

$$\int \frac{d^2u}{dx^2}\frac{du}{dx}\,dx = \frac{1}{2}\left(\frac{du}{dx}\right)^2 \qquad ④$$

等のU(x)の関数形に依存しない積分ルールを対話的にシステムに組み込むことを考えてみたい。

```
MAKERULE INTEGRATE F(x)*G(x) ;
    PRM  F,G,X;
    CONDITION
      X=INDET,
      DIFFCLASS(F,X)=DIFFCLASS(G,X)+1,
      FNAME(F)=FNAME(G);
    RETURN  G(x)  2/2;
  ENDRULE;
```

⑤

この様にMAKERULE　により積分に関するルールを登録すると、

```
~.(Uxx(x)*Ux(x));
  ------> Ux(x) 2/2
~.(Uxxx(x)*Uxx(x));
  ----> Uxx(x) 2/2
```

と③，④を実現できる。上記の登録は、

$$\int \frac{d^{n+1}u}{dx^{n+1}} \frac{d^{n}u}{dx^{n}} dx = \frac{1}{2}\left(\frac{d^{n}u}{dx^{n}}\right)^{2}$$

なる微係数がn 次の場合のメタルールの登録になっていることが判る。これは新たにパターンマッチング機能を作成し、それを組み込むことに依り得られた。

⑤において、PRM はパラメタ,CONDITION以下の3 行は条件の併用である。DIFFCLASS(F,x) はF(x)の微分の階数を示し、FNAMA(F)はF(x)の名前である。MAKERULEのF(x)*G(x)のF(x)とG(x)は任意の関数にマッチングできるものであり、これが強力な登録機能の起原となっている。

LET は特定の処理が終了した時点で出力に関するマッチングを行う機能であるが、MAKERULEは処理の前段階でのマッチングであり、より深い意味でマッチングを行うことに注意されたい。従って通常は、PROPERTYを書き替えねば得られない様な深いルールが、MAKERULE機能により簡単に登録できることが判る。

ユーザにシステムの知識無しに内部の処理を変更することを許すこの様の機能は、より多くの数学的の実体を扱おうとするICASにおいては欠かせない。この機能を柔軟に使用するためには、上記のDIFCLASSの様なrecognizerを増やしておくことが必要とされる。

まとめ

ICASにおいて、この種のU の具体的な関数の形に依存しないルールが取り扱えるのは、評価を抑止した関数を形式的に扱う機能が設計されていることによる。この様に、紙と鉛筆を用いた場合と同じ表記法とアルゴリスムが使える様に数式処理システムの機能を改善すると、従来の整式をベースとした数式処理システムでは扱いにくかった種々の数学知識が、数式処理システムに持ち込める様になる。これは数式処理シスムの内部処理を一般ユーザにとってひらかれた形で、知識駆動型にする事に他ならない。ユーザは自らの専門知識を用いて自らの専門分野にマッチした形でシステムの内部処理に関与できる。LET を用いたのでは行えない様な構造的な処理知識をシステムに与えて、システムを特化することができる。これは「ユーザフレンドリィな数式処理」を目指す努力ととらえることも出来よう。それをオペレタを用いた散乱理論について行った例として、我々の開発したOAP — Operater Algebra Packageがあるが、これについては他の場所で公表したい。

〔 参考文献 〕

1 . 対馬，『知的な数式処理を目指して1),2) 』，数式処理通信1 —4 ， 2 — 1 ('84)

2 . 対馬，『数式処理システムにおける知的対話環境の改善』，数理解析研講究録551 , 116 , ('85)

3 . M.A.Hussain and B. Noble ,APPLICATION OF MACSYMA TO CALCULATION IN APPLIED MATHEMATICS , GENERAL ELECTRIC ('83)

付録

1図は

$$\frac{d^2u}{dx^2} = f(u, \frac{\partial u}{\partial x}, x)$$

の巾級解数を求めるパッケヂKAI のリストである。これは係数比較法でなく、U,Ux 等が内部的にそのまま記述出来るICASの特徴を生かして代入法により係数を求めているので、処理速度も速い。

2 図は、巾級数よりそのPADE近似を求めるパッケヂPADEのリストである。FOR を用いて記述してあり、かつ形式的にPade近似を求めて、後にそれに具体的な整式としての形を代入するというMACSYMA と同様の扱いが行われていることが理解されよう。

3 図は、ファイルより全体のプログラムを読みこむ形でバッチ 処理を行ったときの $\frac{d^2u}{dx^2}$= Ux+U の処理の例である。

```
TYPE PADE

A>¥
  TYPE PADE.C
%
A: TAYLOR (SIN(x), x, 0, 6);
L: M: 3
%
TIME: TRUE;
ECHO: TRUE;
T: L+M;
CA: GCOEF(A, x, 0, T)$
FOR (I, 0, T,
        PRINT ('CA), PRINT('[), PRINT(I), PRINT(']), SPACES(1), PRINT ('=),
        SPACES (1), PRTMATH (CA[I]), NEWLINE ());
DEFVAR p[], ENDVAR$
DEFVAR q[], ENDVAR$
DISABLE (p,q)$
P: SIGMA (p[n]*x^n, n, 0, L);
Q: SIGMA (q[n]*x^n, n, 0, M);
EX: EXPAND (Q*A - P)$
CEX: GCOEF (EX, x, 0, T)$
FOR (I, 0, T,
        PRINT ('CEX), PRINT('[), PRINT (I), PRINT (']), SPACES (1), PRINT (
        SPACES (1), PRTMATH (CEX[I]), NEWLINE ());
q[0]: 1;
XAR: []$
YAR: []$
FOR (I, 0, M-1,
        XAR[I]: CEX[T-I],
        YAR[I]: q[I+1]);
LINEQN (XAR, YAR);
EVSUB (@, q[0], 1);
SETRES (@);
ENABLE (q)$
FOR (I, 0, M,
        PRINT ('q), PRINT ('[), PRINT (I), PRINT (']), SPACES(1), PRINT ('=
        SPACES (1), PRTMATH (q[I]), NEWLINE ());
FOR (I, M+1, T, q[I]: 0)$
FOR (J, 0, L,
        p[J]: SIGMA (CA[K]*q[J-K], K, 0, J))$
ENABLE (p)$
ANSWER: EVAL (EVAL (P)/EVAL (Q));
ECHO: FALSE$
RDS()$
```

第2図　PADE のリスト

```
>TYPE KAI.A
'UNCTION KAI (EX1, EX2, EX3, EX4, EX5),
  C: [],
  D: [],
  DIFEQ2 (EX1, EX2),
  EVCOEF (EX2, EX3, EX4, EX5),
  SIGMA (C[N]*x^N, N, 0, EX2),
:NDFUN $

'UNCTION DIFEQ2 (EX1, EX2, % local % EX3),
  D[0]: U (x),
  D[1]: Ux (x),
  D[2]: EX3: EX1,
  FOR (I, 3, EX2,
       D[I]: EX3: SET (DIF (EX3,x), Uxx (x), EX1) ),
ENDFUN $

FUNCTION EVCOEF (EX1, EX2, EX3, EX4),
  FOR (I, 0, EX1,
       C[I]:
        EVSUB (EVSUB (EVSUB (D[I], Ux(x), EX4), U(x), EX3), x, EX2)/I! ),
ENDFUN $

RDS ()$
```

第 1 図　　KAI の リスト

第3回 途中までの処理結果

```
A: KAI(Ux(x) + U(x)^2, 8, x0, U0, Ux0);¥
   TIME: TRUE;
time:0 sec
@: TRUE

4? A: KAI (Ux(x) + U() x)^2, 8, x0¥
   ;

4? A:KAI (Ux(x) + U(x)^2, 8, x0, U0, Ux0);
time:11 sec
@: U0 + x Ux0 + x^2 Ux0/2 + x^2 U0^2/2

5? DEFINE F U(x), ENDDEF;
  Function  U (x)  accepted
  No definition U
time:1 sec
@: U (x)

6? EVAL(ENT(4));
time:104 sec
@: U0 + x Ux0 + x^2 Ux0/2 + x^2 U0^2/2 + x^3 U0
Ux0/3 + x^3 Ux0/6 + x^3 U0^2/6 + x^4 U0 Ux0/6 + x^
4 Ux0/24 + x^4 U0^2/24 + x^4 U0^3/12 + x^4 Ux0^2/
12 + x^5 U0 Ux0/20 + x^5 Ux0/120 + x^5 U0^2/120 +
x^5 U0^2 Ux0/12 + x^5 U0^3/30 + x^5 Ux0^2/15 + x^6
 U0 Ux0/90 + x^6 U0 Ux0^2/36 + x^6 Ux0/720 + x^6
U0^2/720 + 19/360 x^6 U0^2 Ux0 + x^6 U0^3/120 + x^
6 U0^4/72 + 11/360 x^6 Ux0^2 + x^7 U0 Ux0/504 + 29
/1260 x^7 U0 Ux0^2 + x^7 Ux0/5040 + x^7 U0^2/5040
+ 5/252 x^7 U0^2 Ux0 + x^7 U0^3/630 + x^7 U0^3 Ux0
/63 + 19/2520 x^7 U0^4 + 13/1260 x^7 Ux0^2 + x^7
Ux0^3/252 + x^8 U0 Ux0/3360 + 3/280 x^8 U0 Ux0^2 +
 x^8 Ux0/40320 + x^8 U0^2/40320 + 19/3360 x^8 U0^2
 Ux0 + 5/672 x^8 U0^2 Ux0^2 + x^8 U0^3/4032 + 29/
2520 x^8 U0^3 Ux0 + 5/2016 x^8 U0^4 + x^8 U0^5/504
 + 19/6720 x^8 Ux0^2 + 11/2520 x^8 Ux0^3

7? L: M: 4;
time:0 sec
@: 4

0? RDS('PADE,'C);
time:1 sec
@: PADE
time:0 sec
@: TRUE
time:0 sec
@: TRUE
```

```
3?
T: L+M;
time:0 sec
@: 8

4?
CA: GCOEF(A, x, 0, T)$
time:28 sec

4?
FOR (I, 0, T,
        PRINT ('CA), PRINT('[), PRINT(I), PRINT(']), SPACE
S(1), PRINT ('=),
        SPACES (1), PRTMATH (CA[I]), NEWLINE ());
CA[0] = U0
CA[1] = Ux0
CA[2] = Ux0/2 + U0^2/2
CA[3] = Ux0/6 + U0 Ux0/3 + U0^2/6
CA[4] = Ux0/24 + U0 Ux0/6 + U0^2/24 + U0^3/12 +
Ux0^2/12
CA[5] = Ux0/120 + U0 Ux0/20 + U0^2 Ux0/12 + U0^2/
120 + U0^3/30 + Ux0^2/15
CA[6] = Ux0/720 + U0 Ux0/90 + U0 Ux0^2/36 + 19/360
 U0^2 Ux0 + U0^2/720 + U0^3/120 + U0^4/72 + 11/360
 Ux0^2
CA[7] = Ux0/5040 + U0 Ux0/504 + 29/1260 U0 Ux0^2 +
 5/252 U0^2 Ux0 + U0^3 Ux0/63 + U0^2/5040 + U0^3/
630 + 19/2520 U0^4 + 13/1260 Ux0^2 + Ux0^3/252
CA[8] = Ux0/40320 + U0 Ux0/3360 + 3/280 U0 Ux0^2 +
 19/3360 U0^2 Ux0 + 5/672 U0^2 Ux0^2 + 29/2520 U0^
3 Ux0 + U0^2/40320 + U0^3/4032 + 5/2016 U0^4 + U0^
5/504 + 19/6720 Ux0^2 + 11/2520 Ux0^3
time:11 sec
@: TRUE

5?
DEFVAR p[], ENDVAR$
time:0 sec

5?
DEFVAR q[], ENDVAR$
time:0 sec

5?
DISABLE (p,q)$
time:0 sec

5?
P: SIGMA (p[n]*x^n, n, 0, L);
time:0 sec
@: x p[1] + x^2 p[2] + x^3 p[3] + x^4 p[4] + p[0]

6?
Q: SIGMA (q[n]*x^n, n, 0, M);
time:0 sec
@: x q[1] + x^2 q[2] + x^3 q[3] + x^4 q[4] + q[0]

7?
EX: EXPAND (Q*A - P)$
time:790 sec

7?
CEX: GCOEF (EX, x, 0, T)$
time:174 sec
```

```
7?
FOR (I, 0, T,
        PRINT ('CEX), PRINT('[), PRINT (I), PRINT (']), SP
ACES (1), PRINT ('=),
        SPACES (1), PRTMATH (CEX[I]), NEWLINE ());
CEX[0] = U0 q[0] - p[0]
CEX[1] = U0 q[1] + Ux0 q[0] - p[1]
CEX[2] = U0 q[2] + Ux0 q[0]/2 + Ux0 q[1] + U0^2 q[
0]/2 - p[2]
CEX[3] = U0 Ux0 q[0]/3 + U0 q[3] + Ux0 q[0]/6 +
Ux0 q[1]/2 + Ux0 q[2] + U0^2 q[0]/6 + U0^2 q[1]/2
- p[3]
CEX[4] = U0 Ux0 q[0]/6 + U0 Ux0 q[1]/3 + U0 q[4] +
 Ux0 q[0]/24 + Ux0 q[1]/6 + Ux0 q[2]/2 + Ux0 q[3]
+ U0^2 q[0]/24 + U0^2 q[1]/6 + U0^2 q[2]/2 + U0^3
q[0]/12 + Ux0^2 q[0]/12 - p[4]
CEX[5] = U0 Ux0 q[0]/20 + U0 Ux0 q[1]/6 + U0 Ux0 q
[2]/3 + Ux0 q[0]/120 + Ux0 q[1]/24 + Ux0 q[2]/6 +
Ux0 q[3]/2 + Ux0 q[4] + U0^2 Ux0 q[0]/12 + U0^2 q[
0]/120 + U0^2 q[1]/24 + U0^2 q[2]/6 + U0^2 q[3]/2
+ U0^3 q[0]/30 + U0^3 q[1]/12 + Ux0^2 q[0]/15 +
Ux0^2 q[1]/12
CEX[6] = U0 Ux0 q[0]/90 + U0 Ux0 q[1]/20 + U0 Ux0
q[2]/6 + U0 Ux0 q[3]/3 + U0 Ux0^2 q[0]/36 + Ux0 q[
0]/720 + Ux0 q[1]/120 + Ux0 q[2]/24 + Ux0 q[3]/6 +
 Ux0 q[4]/2 + 19/360 U0^2 Ux0 q[0] + U0^2 Ux0 q[1]
/12 + U0^2 q[0]/720 + U0^2 q[1]/120 + U0^2 q[2]/24
 + U0^2 q[3]/6 + U0^2 q[4]/2 + U0^3 q[0]/120 + U0^
3 q[1]/30 + U0^3 q[2]/12 + U0^4 q[0]/72 + 11/360
Ux0^2 q[0] + Ux0^2 q[1]/15 + Ux0^2 q[2]/12
CEX[7] = U0 Ux0 q[0]/504 + U0 Ux0 q[1]/90 + U0 Ux0
 q[2]/20 + U0 Ux0 q[3]/6 + U0 Ux0 q[4]/3 + 29/1260
 U0 Ux0^2 q[0] + U0 Ux0^2 q[1]/36 + Ux0 q[0]/5040
+ Ux0 q[1]/720 + Ux0 q[2]/120 + Ux0 q[3]/24 + Ux0
q[4]/6 + 5/252 U0^2 Ux0 q[0] + 19/360 U0^2 Ux0 q[1
] + U0^2 Ux0 q[2]/12 + U0^2 q[0]/5040 + U0^2 q[1]/
720 + U0^2 q[2]/120 + U0^2 q[3]/24 + U0^2 q[4]/6 +
 U0^3 Ux0 q[0]/63 + U0^3 q[0]/630 + U0^3 q[1]/120
+ U0^3 q[2]/30 + U0^3 q[3]/12 + 19/2520 U0^4 q[0]
+ U0^4 q[1]/72 + 13/1260 Ux0^2 q[0] + 11/360 Ux0^2
 q[1] + Ux0^2 q[2]/15 + Ux0^2 q[3]/12 + Ux0^3 q[0]
/252
CEX[8] = U0 Ux0 q[0]/3360 + U0 Ux0 q[1]/504 + U0
Ux0 q[2]/90 + U0 Ux0 q[3]/20 + U0 Ux0 q[4]/6 + 3/
280 U0 Ux0^2 q[0] + 29/1260 U0 Ux0^2 q[1] + U0 Ux0
^2 q[2]/36 + Ux0 q[0]/40320 + Ux0 q[1]/5040 + Ux0
q[2]/720 + Ux0 q[3]/120 + Ux0 q[4]/24 + 19/3360 U0
^2 Ux0 q[0] + 5/252 U0^2 Ux0 q[1] + 19/360 U0^2
Ux0 q[2] + U0^2 Ux0 q[3]/12 + 5/672 U0^2 Ux0^2 q[0
] + U0^2 q[0]/40320 + U0^2 q[1]/5040 + U0^2 q[2]/
720 + U0^2 q[3]/120 + U0^2 q[4]/24 + 29/2520 U0^3
Ux0 q[0] + U0^3 Ux0 q[1]/63 + U0^3 q[0]/4032 + U0^
3 q[1]/630 + U0^3 q[2]/120 + U0^3 q[3]/30 + U0^3 q
[4]/12 + 5/2016 U0^4 q[0] + 19/2520 U0^4 q[1] + U0
^4 q[2]/72 + U0^5 q[0]/504 + 19/6720 Ux0^2 q[0] +
13/1260 Ux0^2 q[1] + 11/360 Ux0^2 q[2] + Ux0^2 q[3
]/15 + Ux0^2 q[4]/12 + 11/2520 Ux0^3 q[0] + Ux0^3
q[1]/252
time:71 sec
@: TRUE

0?
q[0]: 1;
time:0 sec
@: q
```