1. Publicação nº	2. Versão	3. Data	5. Distribuição
INPE-2980-PRE/442		*Dez., 1983*	☐ Interna ☒ Externa ☐ Restrita

4. Origem: *DIN/DPD* — Programa: *DENUME/INFORMÁTICA*

6. Palavras chaves - selecionadas pelo(s) autor(es)
SÉRIES DE POTÊNCIA
MANIPULAÇÃO SIMBÓLICA
SOLUÇÃO DE EQUAÇÕES DIFERENCIAIS
REDUCE 2

7. C.D.U.: *519.62*

8. Título *INPE-2980-PRE/442*

SOLUÇÃO ANALÍTICA POR COMPUTADORES DE EQUAÇÕES DIFERENCIAIS ORDINÁRIAS DE PRIMEIRA E SEGUNDA ORDENS UTILIZANDO O MÉTODO DE DESENVOLVIMENTO EM SÉRIES.

10. Páginas: *18*

11. Última página: *A.3*

12. Revisada por

José A.G. Pereira

9. Autoria *Guilherme Bittencourt*
Luiz Alberto Vieira Dias

Assinatura responsável Bittencourt

13. Autorizada por

Parada

Nelson de Jesus Parada
Diretor Geral

14. Resumo/Notas *Neste trabalho, com base na capacidade de Manipulação simbólica do programa "Reduce 2", desenvolveu-se um conjunto de procedimentos com as seguintes funções: interpretar uma equação diferencial escrita em notação semelhante à usual em Matemática e, seguindo o método de Frobenius, clacular uma solução em série de potências para esta equação, caso ela seja linear. O acesso ao sistema pode ser interativo ou através de "Batch". Mansagens de erro e/ou de acompanhamento da solução são emitidas caso necessário. O presente trabalho complementa o estudo anterior, desenvolvido pelos autores, de um solucionador analítico de equações diferenciais ordinárias por métodos heurísticos.*

15. Observações

Este trabalho foi apresentado no 6º Congresso Nacional de Matemática Aplicada e Computacional, em São José dos Campos.

ABSTRACT

Based on "Reduce 2" symbolic manipulation capacities, a set of procedures was developed and implemented, in order analytically solve differencial equations. The input equation is written in the usual mathematical form, and if it is linear, the frovenius method is used, yeilding a power series solution. The system may be interactive or on batch. Errors and/or follow-up massages are displayed, if necessary. This work complements a previous study by the authors on analytical ordinary differencial equations solution by computer, using heuristic methods.

1. INTRODUÇÃO

Descreve-se sucintamente a estrutura do manipulador simbólico "REDUCE 2", dando mais ênfase à capacidade de manipulação e de programação do sistema do que à parte de operação. A seguir, apresenta-se o sistema para manipulação de séries infinitas no "REDUCE 2", desenvolvido para servir de base ao solucionador de equações diferenciais por séries, e finalmente descreve-se o solucionador propriamente dito.

No apêndice apresentam-se exemplos de operação do solucionador de equações diferenciais.

2. O SISTEMA "REDUCE 2"

O sistema de manipulação simbólica "REDUCE 2" é composto de quatro módulos principais (Figura 1).

2.1 - MÓDULO "LISP"

Este módulo consiste em um interpretador da linguagem "LISP", e serve de base para os demais módulos escritos nesta linguagem. Neste módulo estão implementadas, também, as interfaces para os vários periféricos específicos de cada instalação.

No sistema utilizado no presente trabalho o módulo "LISP" apresenta uma limitação que deve ser mencionada: não são admitidos números em ponto flutuante, apenas inteiros e números racionais. Esta limitação não é inerente ao sistema "REDUCE 2", mas apenas ao sistema específico que foi utilizado.

2.2 - MÓDULO SIMBÓLICO

Este módulo consiste em um tradutor da sintaxe "LISP" para a sintaxe "RLISP", isto é, quando se está operando em modo simbólico po dem-se escrever programas para o "LISP" do módulo 1, utilizando a sintaxe "ALGOL-LIKE" do "RLISP".

A grande utilidade deste módulo reside no fato de que os procedimentos escritos em "RLISP" podem ser declarados "OPERATOR" e utilizados a partir do modo algébrico, oferecendo uma possibilidade de expansão da capacidade de manipulação simbólica do sistema.

2.3 - INTEGRADOR

Este módulo é um programa escrito em "LISP" com a finalidade de realizar a integração simbólica de funções, utilizando o algoritmo de Risch. A interação com o usuário dá-se através do operador "INT (função, variável)" a partir do modo algébrico.

2.4 - MÓDULO ALGÉBRICO

Trata-se do manipulador simbólico propriamente dito; ao ser rodado o programa "REDUCE 2" o controle inicia-se neste módulo. Quando se quer passar do modo algébrico para o simbólico deve-se digitar "symbolic"; para o "LISP", "END"; para voltar do simbólico para o algébrico, "ALGEBRAIC" e para o "LISP", ?BEGIN.

Este módulo consiste em um interpretador para uma linguagem de alto nível muito semelhante ao "ALGOL", com a particularidade de que as variáveis podem assumir tanto valores numéricos como simbólicos. As operações matemáticas usuais estão presentes, bem como algumas funções básicas (sin, cos, log, etc.).

Além disto estão presentes algumas funções específicas para manipulação simbólica, como por exemplo a função "DF (expressão, variável, ordem)" que calcula a derivada,em relação a uma dada variável, de qualquer ordem da expressão dada, ou a função "COEFF" que permite obter o coeficiente de uma determinada variável numa dada potência dentro de uma expressão.

No entanto, a flexibilidade maior do sistema consiste na possibilidade de expansão destas capacidades de manipulação através da definição de operadores no modo simbólico, como já foi previamente comentado.

3. MANIPULAÇÃO DE SÉRIES

Como o programa "REDUCE 2" não conta com a capacidade de manipulação de séries infinitas foi necessário escrever módulos para permitir tal manipulação.

Estes módulos foram criados de uma maneira hierárquica de modo que o usuário pode inicialmente utilizar apenas o módulo básico e a medida que a manipulação desejada se tornar mais sofisticada os outros módulos irão se tornando necessários. A seguir são descritos os vários módulos.

3.1 - MÓDULO BÁSICO

Fazem parte deste módulo as rotinas "SOMA", "EXECUTE", "SERIE", "FAT" e "FFFE".

A rotina "SOMA" é usada para definir os somatórios; sua sintaxe é a seguinte:

SOMA (<termo geral>, <índice>, <início>, <fim>);

A rotina verifica se algum dos limites se encontra indefinido e se for o caso devolve a expressão:

SOMATORIO (<ter geral>, <ind>, <início>, <fim>);

caso contrário, a rotina realiza a soma e devolve o valor.

A rotina "EXECUTE" serve para executar um somatório pré-definido no caso deste ter seus limites modificados; sua sintaxe é:

EXECUTE (<somatório>).

A rotina "SERIE" fornece as séries equivalentes às funções básicas (sin, cos, tan, exp, bessel, etc.); sua sintaxe é:

SERIE (<função> (<argumento>));

Além destas, foram implementadas duas rotinas auxiliares para a definição das funções "FATORIAL" e "FATORIAL EXTENDIDO":

FAT (N);

Onde é devolvida a expressão fatorial (N) se N for indefinido, caso contrário o fatorial do número é devolvido:

FFFE (E, N);

em que é devolvida a expressão "FFF (E, N)" caso "E" ou "N" não sejam numéricos, caso contrário é devolvido o valor: E*(E+1)*(E+2)*...*(E+N-1).

3.2 - ROTINAS AUXILIARES

Este módulo é composto de rotinas gerais que aumentam a capacidade de manipulação simbólica do "REDUCE 2", tais como

LIVRE - informa se uma expressão dada é livre de uma determinada variável.

PARCELA - devolve a primeira parcela de uma soma

RESTO - devolve uma soma sem a primeira parcela ou zero se só houver uma parcela.

ELE - devolve o enésimo elemento de uma lista e pode ser usada para obter os parâmetros do somatório.

RAIZ - devolve a raiz enésima de um valor qualquer, deixando indicada a raiz caso o valor não seja numérico. Aceita números negativos, frações e expoentes não-numéricos.

Além destas existem ainda duas rotinas auxiliares específicas para a manipulação de séries:

COMPATÍVEL - informa se duas séries têm os mesmos limites.

UNIFICADO - informa se uma expressão é composta apenas de um somatório ou se é uma expressão complexa.

3.3 - MÓDULO PARA CÁLCULOS COM SÉRIES

Este módulo deve ser usado caso o usuário deseje calcular expressões utilizando séries para obter um só somatório como resultado.

Basicamente este módulo é formado pela rotina "CALCULAR", cuja sintaxe é:

CALCULAR (<expressão>);

em que <expressão> é uma expressão aritmética (sem divisões!), onde os elementos podem ser séries, variáveis ou números, e como resultado é devolvido um único somatório, caso isto seja possível.

As demais rotinas do módulo são invocadas a partir da rotina "CALCULAR" e são transparentes ao usuário.

3.4 - MÓDULO DE FUNÇÕES ESPECIAIS

Neste módulo estão as funções mais elaboradas, tais como:

a) DERIVAR (<expressão>, <var>, <num>),

cuja função é derivar a <expressão> em relação à variável <var> <num> vezes; os somatórios são devidamente tratados como tal.

b) ESCREVER (<somatório>),

cuja função é imprimir de uma maneira "elegante" o <somatório>. Por exemplo, SOMATORIO (3*X**N,N,0,INF), será impresso como:

```
   INF
\ =====
          N
   >  3*X

 /=====
  N = 0
```

4. SOLUCIONADOR DE EQUAÇÕES DIFERENCIAIS

4.1 - O MÉTODO DE FROBENIUS

Este método afirma que, dada uma equação diferencial homogênea, linear de segunda ordem, da forma:

Y'' + P(X) Y' + Q(X) Y = 0

onde P e Q são polinômios em X, então em um ponto ordinário qualquer, X = XP, a solução existe e é da forma:

Y = SOMATORIO (A(N)*X**N, N,0,INF),

onde A(0) e A(1) são constantes arbitrárias.

O método afirma ainda que para um ponto singular regular, X = XS, a solução é a soma de duas soluções linearmente independentes:

```
Y = K1 * SOMATORIO (AN(R1)*X**(R1+N), N,0,INF) +
    K2 * SOMATORIO (BN(R2)*X**(R2+N), N,0,INF)
```

onde R1 e R2 são os expoentes da singularidade.

Existem dois casos especiais de valores de R1 e R2:

1) R1 = R2:

Neste caso os BNs são dados pela derivada de AN em relação a R, calculada no ponto R_1.

2) R1 - R2 = número inteiro:

Neste caso os BNs são dados pela derivada de (R-R2)*AN em relação a R, calculada no ponto R2; e a segunda solução contém um termo logarítmico, exceto para o caso muito especial em que um outro AN é arbitrário além do AO.

4.2 - USO DO SISTEMA

As rotinas para a solução de equações diferenciais por série estão baseadas no "Módulo Básico" para manipulação de séries e em algumas rotinas do módulo auxiliar, a saber, "ELE", "LIVRE", "RAIZ" e "ESCREVER".

Para acessar o sistema o usuário dispõe de duas maneiras:

1) com interpretação da equação:

Neste modo o usuário define sua equação de uma maneira geral (observando que Y'' = DIF(u, X, 2) e Y' = DIF(Y, X) e após entrar as rotinas do solucionador com o comando:

```
IN SEDSERINT;
```

digita o comando:

```
SEDSER (E, Y, X);
```

onde E é a equação, Y a variável dependente e X a variável independente.

2) sem interpretação da equação:

Neste modo o usuário entra as rotinas do solucionador digitando:

```
IN SEDSERBAT;
```

e após digita:

```
SEDSER (A1, A2, A3, A4, Y, X);
```

onde A1, A2, A3 e A4 são os coeficientes da equação, Y é a variável dependente e X a variável independente.

4.3 - OPERAÇÃO DO SISTEMA

O sistema consiste em uma série de rotinas que implementam o método de Frobenius.

Inicialmente a equação é classificada quanto à linearidade e ordem. A seguir são calculadas as raízes da equação indicial:

```
R**2 + (PO-1) R + QO = 0
```

onde PO e QO são dados pelos seguintes:

```
PO = LIM (X P(X))
      X -> 0
```

```
QO = LIM (X**2 Q(X))
     X -> 0
```

Uma vez obtidas as raízes calcula-se a fórmula de recorrência para os ANs, resultante da substituição de:

```
Y = SOMATORIO (A(N)*X**(N+R),N,0,INF);
```

na equação original. A fórmula de recorrência fica disponível ao usuário através da variável "RR" e em geral é uma função do tipo:

```
A(N) = SOMATORIO (FI(N,R)*A(N-I),I,1,N);
```

No caso de A(N) depender de apenas um A(N-M) então calcula-se a fórmula de recorrência fechada (em função de A(0)) que fica disponível para o usuário na variável "RRF".

Caso tenha sido possível calcular a fórmula de recorrência fechada passa-se ao cálculo das soluções propriamente ditas. Para os casos especiais referidos acima foram desenvolvidos procedimentos especiais para a derivação da fórmula de recorrência, que em geral envolve funções fatoriais expandidas ("FFF").

Finalmente, caso o usuário tenha inicializado a variável "EXPANDIR" com o valor "SIM" e a variável "NTERMO" com um valor inteiro as soluções serão expandidas até o termo "NTERMO". Esta última opção é especialmente útil no caos de não ter sido possível calcular a fórmula de recorrência fechada e, por conseguinte, não se dispor de expressões fechadas para as soluções.

A saída do programa consiste em uma listagem com: equação diferencial, classificação da equação, raízes da equação indicial, fórmula de recorrência, fórmula de recorrência fechada (se possível), soluções completas (se possível) e soluções expandidas (se solicitado).

5. EXPANSÕES

Como expansões ao presente sistema propõem-se algoritmos específicos para construir fórmulas de recorrências fechadas a partir de relações de recorrência que contenham em sua fórmula mais de um coeficiente anterior, bem como algoritmos para determinar se as soluções encontradas são do tipo polinomial ou se são realmente séries infinitas.

Algoritmos para cálculo dos raios de convergência das séries também seriam muito úteis e para tal necessita-se de algoritmos gerais para cálculo de limites.

6. CONCLUSÃO

Este sistema foi testado com diversos exemplos extraídos de livros de equações diferenciais de nível de graduação e apresentou resultados bons.

As soluções apresentadas são úteis sob vários aspectos: como material de apoio para estudantes que estejam fazendo cursos de equações diferenciais; para profissionais que desejam a forma da solução ou os primeiros termos de sua expansão em série e não dispõem de muito tempo; ou para uma análise de um grande número de equações num intervalo curto de tempo.

Este sistema foi implementado como mais uma parte de um sistema geral de solução de equações diferenciais, com base no programa de manipulação simbólica "REDUCE 2", que pode também ser extendido, no futuro, para a análise e solução de equações e sistemas de equações gerais e não apenas diferenciais; fazendo do sistema de manipulaão simbólica uma ferramenta de programação poderosa, uma vez que é possível a implementação de interfaces com outras linguagens, voltadas para o cálculo numérico, como "ALGOL", "FOTRAN" ou "PASCAL".

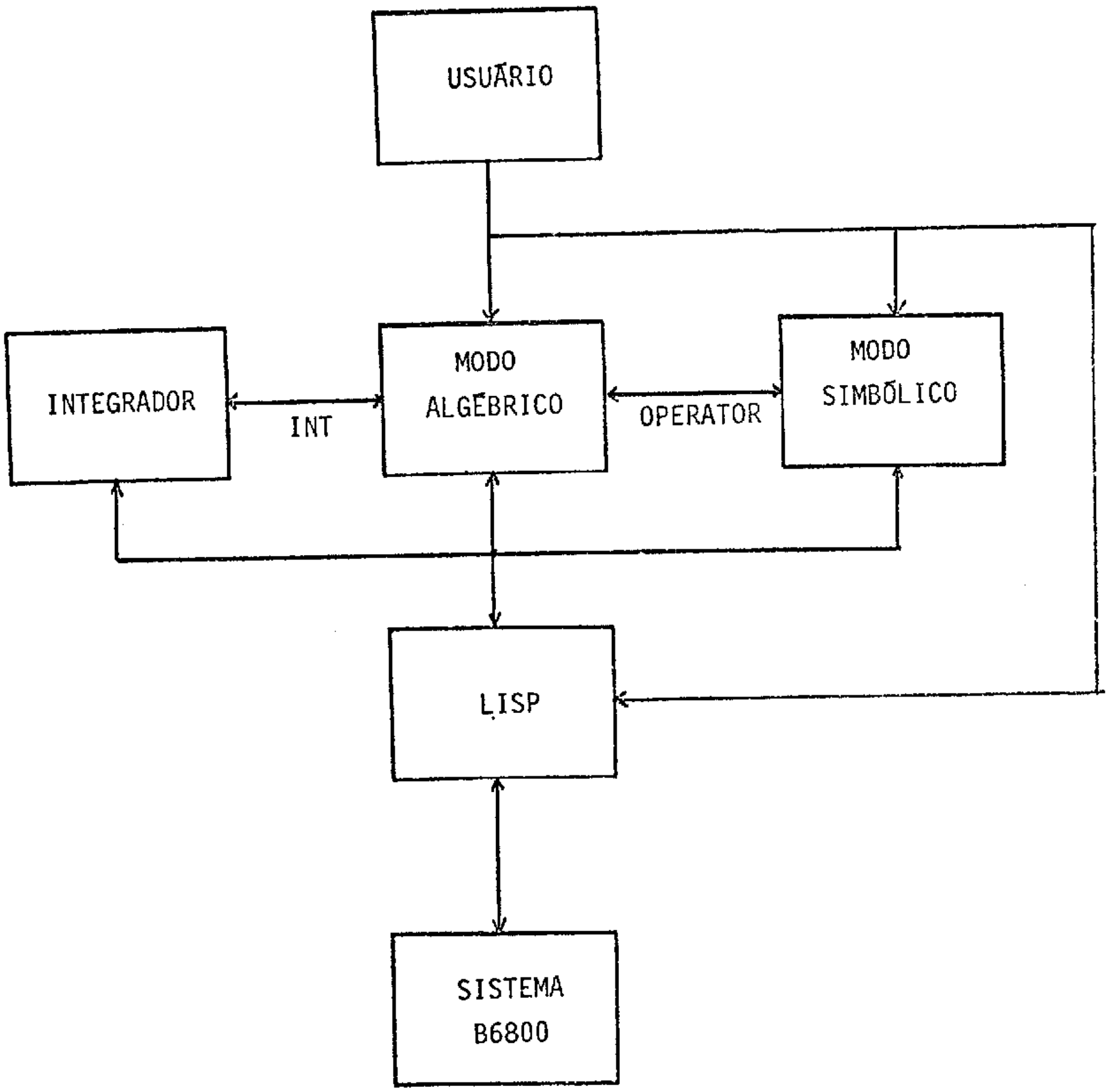

Fig. 1 - Módulos principais do sistema de manipulação simbólica REDUCE 2.

REFERÊNCIAS BIBLIOGRÁFICAS

BOYCE, W.E.; DIPRIMA, R.C. *Elementary differencial equations.* 3 ed. U.S.A. John Wiley & Sons. 1977.

CODDINGTON, E.A. *An introduction to ordinary differencial equations.* Enclewood Cliffs, N.J., Prentice-Hall, Inc., 1961.

HEARN, A.C. *Reduce 2 user's manual.* Salt Lake City, Utah, U.S.A., University of Utah, 1973.

HEARN, A.C. Reduce, a user-oriented interactive system for algebriac simplification. In: *ACM proceedings of the ACM Symposium on Interactive Systems for Experimental Applied Mathematics.* Washington, D.C., *ACM*, 1967.

KELLS, L.M. *Elementary differencial equations.* Mcgraw-Hill Book Company, 1960.

KISELIOV, A.; KRASNOV, M.; MAKARENKO, G. *Problemas de equaciones diferenciales ordinarias.* Moscou, U.R.S.S., Editorial MIR, 1973.

KNUTH, D.E. *The art of computer programming.* Mass., U.S.A., Addison-Wesley, 1973. 4 v.

LAFFERTY, E.L. Power series solutions of ordinary differencial equations on macsyma. In: *Nasa proceedings of the 1977 macsyma user's conference.* Berkeley, California, Nasa scientific and technical information office, 1977. p. 347.

MARTI, J.B.; HEARN, A.C.; GRISS, M.L.; GRISS, C. *Standard lisp report.* Salt Lake City, Utah, University of Utah. 1979.

SPIEGEL, M.R. *Applied differencial equations.* Englewood Cliffs, N.J., Prentice Hall, 1958.

APÊNDICE

```
*  EQUACAO DIFERENCIAL :

        2
( 2*X     )* Y'' +(   - X )* Y' +( X + 1 )* Y = 0

* ORDEM                  : SEGUNDA

* LINEARIDADE            : LINEAR

* METODO                 : SERIES

* SOLUCAO COMPLETA       : SIM


*  RAIZES DA EQUACAO INDICIAL :

R1 = 1

R2 = 1/2

*  FORMULA DE RECORRENCIA :

                                                2
A(N) = A(N - 1)/(3*N + 3*R - 2*(N + R)   - 1)


*  FORMULA DE RECORRENCIA FECHADA :

                N         N
A(N) = (( - 1)  *A(0))/(2  *FFF((2*R + 1)/2,N)*FFF(R,N))


*  SOLUCOES :

       INF

      \=====
              N        N        N
Y1 =   >    ((X *( - 1)  *X)/(2  *FATORIAL(N)*FFF(3/2,N)))

      /=====
       N=0

       INF

      \=====
                ((2*N + 1)/2)          N      N
Y2 =   >    ((X               *( - 1)  )/(2  *FATORIAL(N)*FFF(1/2,N)))

      /=====
       N=0


*  SOLUCAO EXPANDIDA ATE' O 5 TERMO :

                          6             5            4           3         2
Y1 =  - 1/1247400*X  + 1/22680*X  - 1/630*X  + 1/30*X  - 1/3*X  + X

                      (1/2)   5             (1/2)   4           (1/2)   3
Y2 =  - 1/113400*X        *X  + 1/2520*X        *X  - 1/90*X        *X  + 1/6*

      (1/2)   2    (1/2)       (1/2)
     X      *X  - X      *X + X
```

* EQUACAO DIFERENCIAL :

$$(X^{2})*\ Y'' + (3*X)*\ Y' + (X + 1)*\ Y = 0$$

* ORDEM : SEGUNDA

* LINEARIDADE : LINEAR

* METODO : SERIES

* SOLUCAO COMPLETA : SIM

* RAIZES DA EQUACAO INDICIAL :

$R1 = (-1)$

$R2 = (-1)$

* FORMULA DE RECORRENCIA :

$$A(N) = (- A(N - 1))/(2*N + 2*R + (N + R)^{2} + 1)$$

* FORMULA DE RECORRENCIA FECHADA :

$$A(N) = ((- 1)^{N}*A(0))/FFF(R + 2,N)^{2}$$

* SOLUCOES :

$$Y1 = \sum_{N=0}^{INF} ((X^{N}*(- 1)^{N})/(X*FATORIAL(N)^{2}))$$

$$Y2 = SOMATORIO((- 2*X^{N}*(- 1)^{N}*SOMATORIO(1/(K + 1),K,0,N - 1))/(X*FATORIAL(N)^{2}),N,0,INF) + SOMATORIO((X^{N}*(- 1)^{N})/(X*FATORIAL(N)^{2}),N,0,INF)*LOG(X)$$

* SOLUCAO EXPANDIDA ATE' O 5 TERMO :

$$Y1 = - 1/14400*X^{4} + 1/576*X^{3} - 1/36*X^{2} + 1/4*X + X^{(-1)} - 1$$

$$Y2 = - 1/14400*X^{4}*LOG(X) + 137/432000*X^{4} + 1/576*X^{3}*LOG(X) - 25/3456*X^{3} - 1/36*X^{2}*LOG(X) + 11/108*X^{2} + 1/4*X*LOG(X) - 3/4*X + X^{(-1)}*LOG(X) - LOG(X) + 2$$

```
*  EQUACAO DIFERENCIAL :

     2              2
( X   )* Y'' +( 2*X   )* Y' +( (-2) )* Y = 0

* ORDEM                 : SEGUNDA

* LINEARIDADE           : LINEAR

* METODO                : SERIES

* SOLUCAO COMPLETA      : SIM


*  RAIZES DA EQUACAO INDICIAL :

R1 = 2

R2 = (-1)


*  FORMULA DE RECORRENCIA :

                                                    2
A(N) = (2*(N + R - 1)*A(N - 1))/(N + R - (N + R)   + 2)


*  FORMULA DE RECORRENCIA FECHADA :

                N
A(N) = (( - 2)  *FFF(R,N)*A(0))/(FFF(R - 1,N)*FFF(R + 2,N))


*  SOLUCOES :

        INF
       \=====         N        N  2
Y1 =    >     ((X  *( - 2)  *X  *FATORIAL(N + 1))/(FFF(4,N)*FATORIAL(N)))
       /=====
        N=0

        INF
       \=====         N       N
Y2 =    >     ((X  *( - 2)  *(2*SOMATORIO(1/(K - 1),K,0,N - 1) - 2*SOMATORIO(1/(K +
                  1),K,0,N - 4) - 2*SOMATORIO(1/(K + 1),K,0,N - 1) + 3)*FFF((
                  -1),N))/(4*X*FATORIAL(N)*FATORIAL(N - 3)))
       /=====
        N=0
```