

OBJETIVOS DO CAPÍTULO

- Calcular o resultado de séries infinitas
- Aplicar os métodos de iteração linear, bisseção e Newton para determinar raízes de equações
- Revisar o uso de arquivos de dados e de resultados, Notepad e Wgnuplot
- Alterar dinamicamente arquivo de comandos do programa de gráficos Wgnuplot e usar novos comandos

Para inicializar as atividades deste capítulo, deve-se acessar o programa Fortran, no Windows, através de: **Start, Programs, Fortran PowerStation 4.0, Microsoft Developer Studio**

18.1 programa18a.f90

- 1) Objetivos do programa:
 - (a) exemplificar o cálculo do resultado de séries infinitas;
 - (b) revisar o uso de arquivos de dados e de resultados, Notepad e Wgnuplot; e
 - (c) alterar dinamicamente arquivo de comandos do programa de gráficos Wgnuplot e usar novos comandos.
- 2) No Fortran, seguindo o [procedimento-padrão](#), **criar um projeto** com o nome **projeto_18**
- 3) **Acessar** o site ftp://ftp.demec.ufpr.br/Disciplinas/Tm784/projeto_18
- 4) **Clicar** com o botão do lado direito do mouse sobre o arquivo programa18a.f90
- 5) **Escolher** a opção Copiar para pasta... (Save Target As)
- 6) **Localizar** a pasta do projeto
- 7) **Clicar** no botão OK
- 8) **Repetir** os itens 3 a 7, acima, para os arquivos dados18a.txt, comandos18a.gnu, Wgnuplot.exe
- 9) No Fortran, seguindo o [procedimento-padrão](#), **inserir** no projeto o programa-fonte **programa18a.f90**, mostrado na **Tabela 18.1**.
- 10) **Estudar o programa-principal** considerando os comentários do item 11, abaixo.
- 11) Comentários sobre o programa:
 - (a) O programa18a.f90 é composto pelo programa-principal e seis sub-rotinas.
 - (b) A sub-rotina DADOS é usada para ler os dados do programa do arquivo dados18a.txt.

- (c) A sub-rotina `SERIE_1` é usada para calcular a soma dos N termos da seguinte série geométrica infinita:

$$S = \frac{1}{2} + \frac{1}{4} + \frac{1}{8} + \dots = \sum_{l=1}^{\infty} \left(\frac{1}{2^l} \right) = 1 \quad (18.1)$$

- (d) A sub-rotina `SERIE_2` é usada para calcular o valor de π através da soma dos N termos da seguinte série infinita:

$$S = 4 \left(1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \frac{1}{9} - \dots \right) = 4 \sum_{l=0}^{\infty} \left[\frac{(-1)^l}{(2l+1)} \right] = \pi \quad (18.2)$$

- (e) A sub-rotina `SERIE_3` é usada para calcular o valor do número e através da soma dos N termos da seguinte série infinita, onde cada termo envolve a sub-rotina `FATORIAL`:

$$S = 1 + \frac{1}{1!} + \frac{1}{2!} + \frac{1}{3!} + \dots = \sum_{l=0}^{\infty} \left(\frac{1}{l!} \right) = e \quad (18.3)$$

- (f) As sub-rotinas das três séries também calculam o erro para a solução exata e escrevem no arquivo `saida18a.txt` o número e valor de cada termo da série, a soma dos termos e o erro.
- (g) A sub-rotina `GRAFICO`: abre o arquivo `comandos18a.gnu`; pula as sete primeiras linhas deste arquivo; na oitava linha, escreve um comentário para representar o título do gráfico de acordo com a série escolhida para cálculo; na nona linha, escreve outro comentário, que é o comando `replot` do `Wgnuplot`; finalmente, esta sub-rotina executa o programa `Wgnuplot` para fazer o gráfico semilog do erro do cálculo da série escolhida, em função do número de termos N usado.
- (h) O programa-principal: chama a sub-rotina `DADOS`; cria o arquivo `saida18a.txt`; escreve nele um comentário na primeira linha para indicar o significado de cada coluna de resultados; o símbolo `#` é usado para informar ao `Wgnuplot` que a linha é apenas um comentário, e não dados para o gráfico; dependendo do tipo de série escolhida pelo usuário é chamada a sub-rotina adequada; o `Notepad` abre o arquivo de saída; e, é chamada a sub-rotina para fazer o gráfico do erro.

12) Executar **Build, Compile** para compilar o programa.

13) Gerar o programa-executável fazendo **Build, Build**.

14) Executar o programa através de **Build, Execute. Usar, os dados mostrados na Figura 18.1.**

Tabela 18.1 Programa18a.f90

```

program series

use portlib

implicit none
integer :: i, n, ver, tipo_serie
real*8  :: soma, exato, erro, termo

call dados

open(6,file="saida18a.txt")
write(6,10)
10 format("#      i", t17,"termo", t47,"soma", t77,"erro")

select case ( tipo_serie )
  case ( 1 )
    call serie_1
  case ( 2 )
    call serie_2
  case ( 3 )
    call serie_3
end select

close(6)

ver = system ("Notepad saida18a.txt")

call grafico

!-----

contains

!-----

subroutine dados

  ver = system ("Notepad dados18a.txt")

  open(5,file="dados18a.txt")
  read(5,*) tipo_serie

```

```

read(5,*) n
close(5)

end subroutine dados

!-----

subroutine serie_1

! calcula soma de uma série geométrica

exato = 1

soma = 0

do i = 1, n
    termo = 1.0d0 / ( 2.0d0 ** i )
    soma = soma + termo
    erro = dabs(exato - soma)
    write(6,10) i, termo, soma, erro
    10 format( i8, 3(1pe30.15e3) )
end do

end subroutine serie_1

!-----

subroutine serie_2

! calcula soma da série de pi

exato = dacos(-1.0d0)

soma = 0

do i = 0, n
    termo = 4.0d0 * ((-1)**i) / ( 2.0d0 * i + 1 )
    soma = soma + termo
    erro = dabs(exato - soma)
    write(6,10) i, termo, soma, erro
    10 format( i8, 3(1pe30.15e3) )
end do

```

```

end subroutine serie_2

!-----

subroutine serie_3

! calcula soma da série de e

real*8 fatorial_i

exato = dexp(1.0d0)

soma = 0

do i = 0, n
  call fatorial ( i, fatorial_i)
  termo = 1 / fatorial_i
  soma = soma + termo
  erro = dabs(exato - soma)
  write(6,10) i, termo, soma, erro
  10 format( i8, 3(1pe30.15e3) )
end do

end subroutine serie_3

!-----

subroutine fatorial(j,fat)

! calcula o fatorial de j

integer j
real*8 fat, k

fat = 1

do k = 2, j
  fat = fat * k
end do

end subroutine fatorial

!-----

```

```

subroutine grafico

integer k

open(9,file="comandos18a.gnu")
do k = 1, 7
  read(9,*)
end do

select case ( tipo_serie )
  case ( 1 )
    write(9,*) "set title 'série geométrica'"
  case ( 2 )
    write(9,*) "set title 'série de pi'"
  case ( 3 )
    write(9,*) "set title 'série de e'"
end select

write(9,*) "replot"

close(9)

ver = system ("Wgnuplot comandos18a.gnu")

end subroutine grafico

!-----

end program series

```

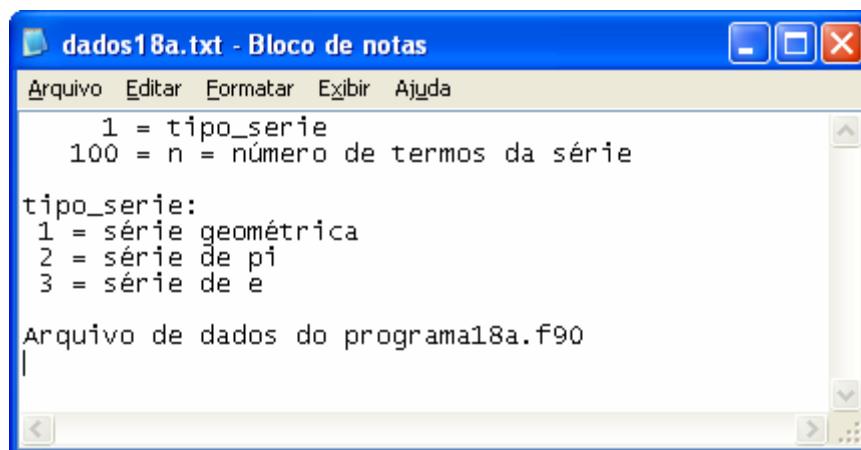


Figura 18.1 Arquivo de dados do programa18a.f90.

- 15) **Analisar os resultados** mostrados nas Figuras 18.2 e 18.3.
- 16) **Executar** novamente o programa usando **tipo_serie = 2** e **analisar** os novos resultados.
- 17) **Executar** novamente o programa usando **tipo_serie = 3** e **analisar** os novos resultados.
- 18) **Executar** novamente o programa usando outros dados e **analisar** os novos resultados.

#	i	termo	soma	erro
1	1	5.000000000000000E-001	5.000000000000000E-001	5.000000000000000E-001
2	2	2.500000000000000E-001	7.500000000000000E-001	2.500000000000000E-001
3	3	1.250000000000000E-001	8.750000000000000E-001	1.250000000000000E-001
4	4	6.250000000000000E-002	9.375000000000000E-001	6.250000000000000E-002
5	5	3.125000000000000E-002	9.687500000000000E-001	3.125000000000000E-002
50	50	8.881784197001252E-016	9.999999999999991E-001	8.881784197001252E-016
51	51	4.440892098500626E-016	9.999999999999996E-001	4.440892098500626E-016
52	52	2.220446049250313E-016	9.999999999999998E-001	2.220446049250313E-016
53	53	1.110223024625157E-016	9.999999999999999E-001	1.110223024625157E-016
54	54	5.551115123125783E-017	1.000000000000000E+000	0.000000000000000E+000
55	55	2.775557561562891E-017	1.000000000000000E+000	0.000000000000000E+000
56	56	1.38778780781446E-017	1.000000000000000E+000	0.000000000000000E+000
99	99	1.577721810442024E-030	1.000000000000000E+000	0.000000000000000E+000
100	100	7.888609052210118E-031	1.000000000000000E+000	0.000000000000000E+000

Figura 18.2 Parte do arquivo de resultados do programa18a.f90 para os dados da Fig. 18.1.

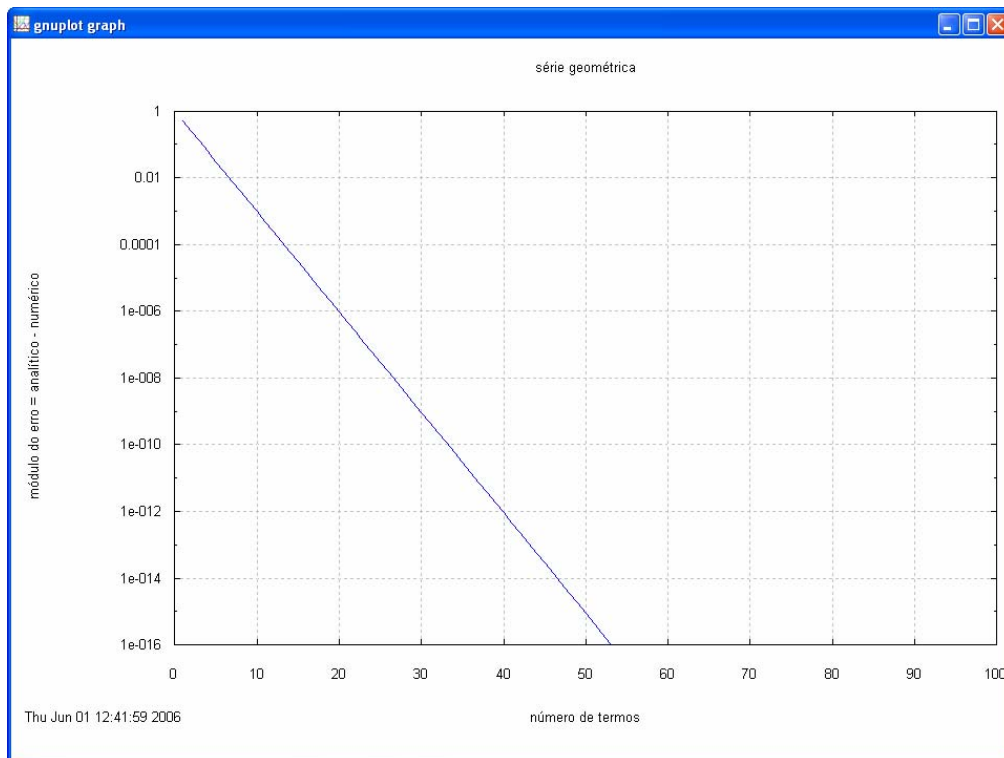


Figura 18.3. Gráfico com resultados do programa18a.f90 para os dados da Fig. 18.1.

18.2 programa18b.f90

- 1) **Objetivo do programa:** para uma equação quadrática específica, aplicar os métodos de iteração linear, bisseção e Newton para determinar suas raízes.
- 2) Nesta seção será usado o mesmo projeto da seção anterior. Portanto, deve-se executar o seguinte no Fortran:
 - a) **Clicar** sobre o nome do programa-fonte.
 - b) **Edit, Cut** para retirar o programa-fonte do projeto.
 - c) **Clicar** dentro do campo de edição de programa-fonte.
 - d) **File, Close.**
- 3) **Acessar** o site ftp://ftp.demec.ufpr.br/Disciplinas/Tm784/projeto_18
- 4) **Clicar** com o botão do lado direito do mouse sobre o arquivo programa18b.f90
- 5) **Escolher** a opção Copiar para pasta... (Save Target As)
- 6) **Localizar** a pasta do projeto
- 7) **Clicar** no botão OK
- 8) **Repetir** os itens 3 a 7, acima, para os arquivos dados18b.txt e comandos18b.gnu
- 9) No Fortran, seguindo o **procedimento-padrão**, **inserir** no projeto o programa-fonte **programa18b.f90**, mostrado na **Tabela 18.2**.
- 10) **Estudar o programa-principal** considerando os comentários do item 11, abaixo.
- 11) Comentários sobre o programa:
 - (a) O programa18b.f90 é composto pelo programa-principal e cinco sub-rotinas.
 - (b) Para as sub-rotinas DADOS e GRAFICO e o programa-principal valem os mesmos comentários da seção anterior, exceto que os arquivos envolvidos são agora dados18b.txt, comandos18b.gnu e saida18b.txt.
 - (c) As sub-rotinas LINEAR, BISSECAO e NEWTON resolvem a equação

$$x^2 - 5x + 6 = 0 \tag{18.4}$$

através dos métodos de iteração linear, bisseção e Newton para determinar suas raízes. Detalhes sobre estes métodos podem ser encontrados na maioria dos livros de cálculo numérico.

- 12) Executar **Build, Compile** para compilar o programa.
- 13) Gerar o programa-executável fazendo **Build, Build**.
- 14) Executar o programa através de **Build, Execute. Usar, os dados mostrados na Figura 18.4**.
- 15) **Analisar os resultados** mostrados nas Figuras 18.5 e 18.6.

- 16) **Executar** novamente o programa usando **tipo_metodo = 2** e **analisar** os novos resultados.
- 17) **Executar** novamente o programa usando **tipo_metodo = 3** e **analisar** os novos resultados.
- 18) **Executar** novamente o programa usando **tipo_metodo = 3**, **n = 20**, **exato = 3** e **xo = 100** e **analisar** os novos resultados.
- 19) **Executar** novamente o programa usando outros dados e **analisar** os novos resultados.
- 20) No Fortran, para fechar o projeto atual, executar **File, Close Workspace**.

Tabela 18.2 Programa18b.f90

```

program raizes

use portlib

implicit none
integer :: i, n, ver, tipo_metodo
real*8  :: exato, erro, xo, x, r

call dados

x = xo

open(6,file="saida18b.txt")
write(6,10)
10 format("#      i", t17,"x", t47,"erro")

erro = dabs(exato - x)
write(6,11) 0, x, erro
11 format( i8, 2(1pe30.15e3) )

select case ( tipo_metodo )
  case ( 1 )
    call linear
  case ( 2 )
    call bissecao
  case ( 3 )
    call newton
end select

close(6)

ver = system ("Notepad saida18b.txt")

```

```

call grafico

!-----

contains

!-----

subroutine dados

  ver = system ("Notepad dados18b.txt")

  open(5,file="dados18b.txt")
  read(5,*) tipo_metodo
  read(5,*) n
  read(5,*) exato
  read(5,*) xo
  read(5,*) r
  close(5)

end subroutine dados

!-----

subroutine linear

  ! calcula raiz de uma equação quadrática com o método da iteração linear

  do i = 1, n
    x = ( ( x ** 2 ) + 6 ) / 5
    erro = dabs(exato - x)
    write(6,10) i, x, erro
    10 format( i8, 2(1pe30.15e3) )
  end do

end subroutine linear

!-----

subroutine bissecao

  ! calcula raiz de uma equação quadrática com o método da bisseção

```

```

real*8 a, b, fa, fx

a = xo - r
b = xo + r

do i = 1, n
  fx = x**2 - 5*x + 6
  fa = a**2 - 5*a + 6
  if ( fx*fa < 0 ) then
    b = x
  else
    a = x
  end if
  x = (a + b) / 2
  erro = dabs(exato - x)
  write(6,10) i, x, erro
  10 format( i8, 2(1pe30.15e3) )
end do

end subroutine bissecao

!-----

subroutine newton

! calcula raiz de uma equação quadrática com o método de Newton

real*8 f, fl

do i = 1, n
  f = x**2 - 5*x + 6
  fl = 2*x - 5
  x = x - f / fl
  erro = dabs(exato - x)
  write(6,10) i, x, erro
  10 format( i8, 2(1pe30.15e3) )
end do

end subroutine newton

!-----

subroutine grafico

```

```

integer k

open(9,file="comandos18b.gnu")
do k = 1, 7
  read(9,*)
end do

select case ( tipo_metodo )
  case ( 1 )
    write(9,*) "set title 'método da iteração linear'"
  case ( 2 )
    write(9,*) "set title 'método da bisseção'"
  case ( 3 )
    write(9,*) "set title 'método de Newton'"
end select

write(9,*) "replot"

close(9)

ver = system ("Wgnuplot comandos18b.gnu")

end subroutine grafico

!-----

end program raizes

```

```

dados18b.txt - Bloco de notas
Arquivo  Editar  Formatar  Exibir  Ajuda
1 = (tipo_metodo)
100 = (n) número de iterações
2 = (exato) solução exata
0 = (xo) estimativa inicial da solução
3 = (r) raio de convergência; só para método da bisseção

tipo_metodo:
1 = linear
2 = bisseção
3 = Newton

Arquivo de dados do programa18b.f90
|

```

Figura 18.4 Arquivo de dados do programa18b.f90.

18.3 EXERCÍCIOS

Exercício 18.1

Adaptar o programa18a.f90 para incluir as seguintes opções de séries:

$$S = \frac{1}{1x2} + \frac{1}{2x3} + \frac{1}{3x4} + \dots = \sum_{I=1}^{\infty} \left[\frac{1}{I(I+1)} \right] = 1 \quad (18.5)$$

$$S = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \frac{x^3}{3!} + \dots = \sum_{I=0}^{\infty} \left(\frac{x^I}{I!} \right) = e^x \quad (18.6)$$

$$S = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \dots = \sum_{I=0}^{\infty} \left[\frac{(-1)^I x^{2I+1}}{(2I+1)!} \right] = \text{sen}(x) \quad (18.7)$$

$$S = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \frac{x^6}{6!} + \dots = \sum_{I=0}^{\infty} \left[\frac{(-1)^I x^{2I}}{(2I)!} \right] = \text{cos}(x) \quad (18.8)$$

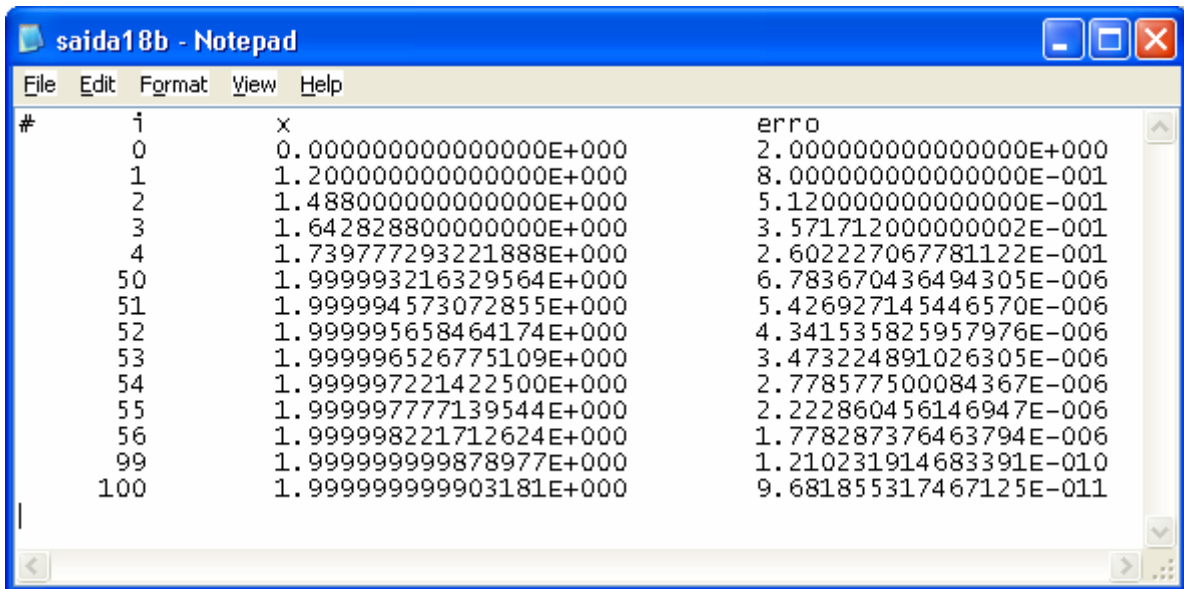


Figura 18.5 Parte do arquivo de resultados do programa18b.f90 para os dados da Fig. 18.4.

Exercício 18.2

Adaptar o programa18a.f90 para usar precisão simples em todos os cálculos com números reais. Comparar os resultados com a versão original do programa18a.f90 na qual usa-se precisão dupla.

Exercício 18.3

Adaptar o programa18b.f90 para usar precisão simples em todos os cálculos com números reais. Comparar os resultados com a versão original do programa18b.f90 na qual usa-se precisão dupla.

Exercício 18.4

Adaptar o programa18b.f90 para resolver a seguinte equação:

$$x^2 - x - 6 = 0 \quad (18.9)$$

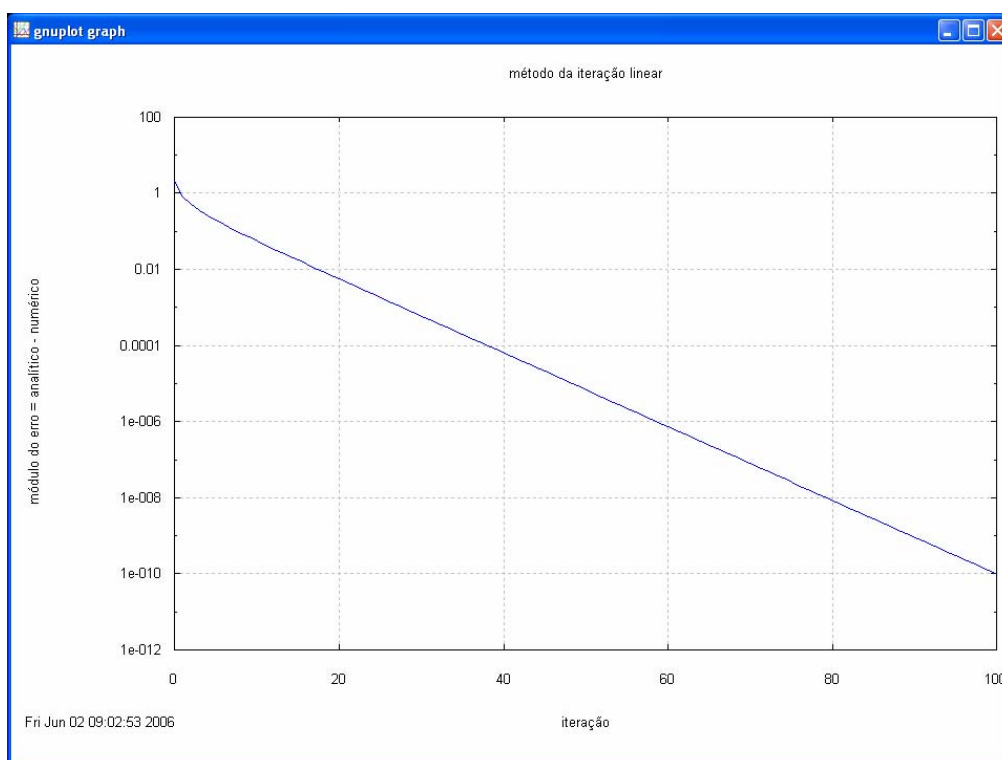


Figura 18.6 Gráfico com resultados do programa18b.f90 para os dados da Fig. 18.4.