

Meteorologia

Apostila de GrADS

Produzida
por
Guilherme Martins - Meteorologista

São José dos Campos - SP
março de 2009

Sumário

1. Introdução	1
2. Instalando o GrADS	1
3. Opções de linhas de comando	2
4. Configurações gerais	2
4.1. <i>Habilitando o slogan do GrADS</i>	2
4.2. <i>Configurando o modo de visualização</i>	2
4.3. <i>O comando “query” ou “q”</i>	3
5. Iniciando o GrADS	5
6. Abrindo arquivos	5
7. Visualizando arquivos	5
8. Variando ou Fixando a latitude e longitude	6
9. Variando o nível de pressão	6
10. Fixando ou variando o tempo	7
11. Tipos de gráfico	7
12. Cores usadas no GrADS	9
13. Estilos de linhas disponíveis no GrADS	9
14. Marcadores disponíveis no GrADS	10
15. Modificando cor, espessura e tamanho dos rótulos dos contornos	10
16. Modificando a espessura da linha do contorno	11
17. Modificando o intervalo dos contornos	11
18. Visualizando valores acima de um determinado valor	12
19. Visualizando valores abaixo de um determinado valor	12
20. Visualizando valores dado um intervalo específico	13
21. Especificando o intervalo de variação do eixo x e y	14
22. Visualizando valores fora de um limite especificado	14
23. Gráfico de Barra	15
24. Preenchendo ou não as barras	15
25. Gráfico de flecha do vento	17
25.1. <i>Comprimento da escala da flecha</i>	17
26. Escrevendo texto no gráfico	17
26.1. <i>Adicionando título</i>	17
26.2. <i>Adicionando título aos eixos x e y</i>	17
27. Invertendo os eixos x e y em um gráfico	18
28. Projeções disponíveis no GrADS	19
29. Invertendo a ordem dos eixos x ou y	20
30. Atribuindo características ao mapa	21
31. Adicionando strings nos rótulos dos contornos	22

32. Fixando valores para o eixo y	22
33. Alterando características dos eixos x e y	23
34. Deslocando o eixo x e/ou y em um gráfico	23
35. Gerando um gráfico de corte vertical com escala logarítmica	25
36. Obtendo as coordenadas x e y de um ponto	25
37. Desenhando um marcador	25
38. Funções intrínsecas do GrADS	25
38.1. Capturando uma linha usando a função "sublin"	25
38.2. Capturando uma palavra usando a função "subwrd"	25
38.3. Capturando parte de uma palavra usando a função "substr"	27
38.4. Obtendo parte de uma palavra usando "substr"	27
39. Desenhando uma caixa no gráfico	28
40. Barra de cores no GrADS	29
41. Salvando figuras no GrADS	30
42. Criando arquivo ".gs" no GrADS	30
43. Executando arquivo ".gs"	31
44. Programas Adicionais	31
44.1. Usando o script grib2ctl.pl	31
44.2. Usando o script lats4d.gs	31
44.3. Procedimentos para instalar EOF no GrADS	33
44.4 Instalando o regrid2 no GrADS	35
45. Linguagem de Programação no GrADS	37
45.1. Operadores	37
45.2. Fluxo de Controle	37
45.2.1. Comando WHILE	37
45.2.2. Comando IF	38
46. Salvando arquivo texto no GrADS	40
47. Aplicações diversas	41
47.1. Gráfico de linha de uma área	41
47.2. Gráfico Hovmoller (x = longitude e y = tempo)	41
47.3. Gráfico Hovmoller (x = latitude e y = altura)	42
47.4. Mascando continente ou oceano	42
47.5. Usando o script "color.gs" para criar escala de cores personalizada	44
47.6. Usando o script "clave.gs" para gerar climatologia	44
47.7. Usando o script "drawbox.gs" para desenhar uma caixa	45
47.8. Usando o script "rmean.gs" para calcular média móvel	45
47.9. Convertendo a Temperatura de Kelvin para Celsius	46
47.10. Configurando os vetores no gráfico de vento	47
47.11. Calculando a vorticidade	48
47.12. Calculando a divergência	49
48. Links interessantes sobre GrADS	49

1. Introdução

Esta apostila destina-se aos usuários que estão iniciando sua pesquisa por meio do pacote gráfico “Grid Análisis and Display System (GrADS)”. A finalidade consiste em dar uma base inicial para quem nunca utilizou este pacote, então não cabe aqui fazer um levantamento sobre sua história, mas sim dar o máximo de informações práticas para que os usuários possam para iniciar seus trabalhos.

Quando iniciamos nossa pesquisa com a utilização do GrADS normalmente nas aulas de Meteorologia Sinótica e Dinâmica, ficamos sem entender praticamente nada, pois são muitos comandos usados. Por isso, esta apostila tem como objetivo ajudar novos usuários bem como servir de material de consulta rápida para possíveis dúvidas.

Divirtam-se.

Guilherme Martins.

2. Instalando o GrADS.

Será usado o GrADS para plataforma Linux. Os arquivos de instalação podem ser baixados do site do GrADS (<http://www.grads.iges.org/grads>). Após baixar os arquivos de instalação, descompacte-os. Crie o diretório “grads” em “/usr/local/lib” (você deve ser “root” para criar o diretório). Todos os arquivos de biblioteca do GrADS (arquivos “.gs”, “.dat” e qualquer script) ficarão neste local. Os arquivos executáveis (gradsc, gradsnc, gribmap, etc) deverão ser copiados para “/usr/local/bin”. Com estes passos o GrADS pode ser usado por qualquer usuário do Linux desde que ele tenha permissão para executar os arquivos.

Para que o GrADS funcione na sua área é necessário adicionar a linha abaixo no arquivo “.bashrc” (que esta no seu home). Para visualizar este arquivo basta digitar no terminal do Linux “ls -a” para ver arquivos ocultos (-a).

Adicione esta linha **export GASCRP=/usr/local/lib/grads**

Não se esqueça de atualizar seu “.bashrc” fazendo “source .bashrc” no terminal do Linux.

3. Opções de linhas de comando

Opções:

-c “comando” → abre o GrADS e executa o “comando”

Exemplo: No terminal do Linux → `gradsnc -c "sdfopen air.mon.nc"`

Isso executará o GrADS e abrirá o arquivo ".nc"

-b → abre o GrADS sem o terminal gráfico.

Exemplo: No terminal do Linux → `gradsnc -b`

-l → abre o GrADS no modo landscape (retrato). O tamanho da janela para esta opção é 11 x 8.5 polegadas.

Exemplo: No terminal do Linux → `gradsnc -l`

-p → abre o GrADS no modo portrait (paisagem) . O tamanho da janela para esta opção é 8.5 x 11 polegadas

Exemplo: No terminal do Linux → `gradsnc -p`

Porém, os comandos podem ser combinados. Veja o exemplo abaixo para fixação.

Exemplo: No terminal do Linux → `gradsnc -lc "script.gs"`

Isso executará o "script.gs" que será aberto no modo landscape.

4. Configurações gerais

4.1. Habilitando o slogan do GrADS

Habilita (`ga-> set grads on`) ou desabilita (`ga-> set grads off`) a propaganda do GrADS (que fica na parte inferior esquerda como também a data que se encontra do lado direito).

4.2. Configurando o modo de visualização

Esta opção configura a janela gráfica do GrADS de acordo com uma cor pré-definida pelo usuário.

Sintaxe : `set display color <cor>`. Onde <cor> pode ser "white" ou "black".

Exemplo:

'reinit'	;*reinicia o GrADS toda vez que o script for executado
'set display color white'	;*fixa a cor branca para janela gráfica do GrADS
'c'	;*limpa a tela para que o comando anterior tenha efeito
'sdfopen vwnd.mon.mean.nc'	;*abre o arquivo ".nc"
'd vwnd'	;*visualiza a variável

4.3. O comando “query” ou “q”

As opções mais interessantes são: config, files, file n, dims, gxinfo, pos, time, clear, reset, reinit, q cltinfo e quit.

Exemplos:

ga-> q cltinfo → fornece detalhes sobre o arquivo aberto

dset air.mon.mean.nc

title Monthly NCEP/DOE Reanalysis 2

undef -9.96921e+36

xdef 144 linear 0 2.5 → número de pontos (144) na direção x, longitude inicial (0) e o incremento (2.5° graus)

ydef 73 linear -90 2.5 → número de pontos (73) na direção y, latitude inicial (-90) e o incremento (2.5° graus)

zdef 17 levels 1000 925 850 700 600 500 400 300 250 200 150 100 70 50 30 20 10 → número de níveis verticais (17) e quais níveis estão presente no seu arquivo

tdef 349 linear 00Z01JAN1979 1mo → número total de tempos (349), primeiro tempo do arquivo (00Z01JAN1979) e seu incremento (1mo = 1mês)

vars 1 → número de variáveis do arquivo

air 17 -999 Monthly Air Temperature on Pressure Levels → nome da variável (air), número de níveis verticais (17), valor tabelado (-999) e uma breve descrição da variável

ga-> q config → fornece informações da versão do GrADS em uso bem como os tipos de arquivos que podem ser abertos

Config: v1.9b4 32-bit little-endian readline sdf/xdx netcdf lats printim

Grid Analysis and Display System (GrADS) Version 1.9b4

Copyright (c) 1988-2005 by Brian Doty and IGES

Center for Ocean-Land-Atmosphere Studies (COLA)

Institute for Global Environment and Society (IGES)

This program is distributed WITHOUT ANY WARRANTY

See file COPYRIGHT for more information.

Built Mon May 23 14:17:20 EDT 2005 for i686-pc-linux-gnu

This version of GrADS has been configured with the following options:

- o This is a 32-bit LITTLE ENDIAN machine version.
- o Command line editing (readline) ENABLED.
- o CIRES/CDC (<http://www.cdc.noaa.gov>) SDF/XDF interface ENABLED.
Use sdfopen/xdxopen to read NetCDF files.
- o DTYPE netcdf is ENABLED; DTYPE hdfsd is DISABLED.
- o OPeNDAP (a.k.a. DODS) gridded data interface DISABLED.
- o OPeNDAP (a.k.a. DODS) station data interface DISABLED.
- o PCMDI (<http://www-pcmdi.llnl.gov>) LATS interface ENABLED.
This version is configured to write GRIB and NetCDF files.
- o DAO (<http://dao.gsfc.nasa.gov>) Athena Widget GUI DISABLED.
- o NRL/DAO/PCMDI XA or ImageMagick Image Output DISABLED.
- o printim command for direct png/gif output ENABLED.
(via the GD Library -- <http://www.boutell.com/gd>)

For additional information please consult <http://grads.iges.org/grads/>

ga-> q files → mostra todos os arquivos abertos

File 1 :

Descriptor: uwnd.1979.2006.ctl

Binary: uwnd.1979.2006.bin

File 2 :

Descriptor: vwnd.1979.2006.ctl

Binary: vwnd.1979.2006.bin

ga-> q file 1 → fornece informações sobre o arquivo que esta aberto no caso de ter aberto 2 ou mais arquivos. O 1 corresponde ao número do arquivo que esta aberto

File 1 :

Descriptor: uwnd.1979.2006.ctl

Binary: uwnd.1979.2006.bin

Type = Gridded

Xsize = 144 Ysize = 73 Zsize = 1 Tsize = 40908

Number of Variables = 1

uwnd 1 33 ** u inv comp-u[m/s]

ga-> q file 2 → fornece informações sobre o arquivo 2 que esta aberto. Tem a mesma aplicação da descrição anterior (q file 1)

File 2 :

Descriptor: vwnd.1979.2006.ctl

Binary: vwnd.1979.2006.bin

Type = Gridded

Xsize = 144 Ysize = 73 Zsize = 1 Tsize = 40908

Number of Variables = 1

vwnd 1 33 ** u inv comp-u[m/s]

ga-> q dims → fornece informações sobre a dimensão do arquivo aberto

Default file number is: 1

X is varying Lon = 0 to 360 X = 1 to 145

Y is varying Lat = -90 to 90 Y = 0.999999 to 73

Z is fixed Lev = 700 Z = 1

T is fixed Time = 00Z01JAN1979 T = 1

ga-> q gxinfo → fornece informações sobre como esta configurada a página, informações sobre os eixos x e y e qual o tipo de projeção usada

Last Graphic = Clear

Page Size = 11 by 8.5

X Limits = 0 to 11

Y Limits = 0 to 8.5

Xaxis = None Yaxis = None

Mproj = 2

ga-> q pos → Obtém as coordenadas da página. Digite o comando e dê um clique com o mouse em alguma área da página que será retornado alguns valores como no resultado abaixo. O valor 4.15556 corresponde ao ponto x e o valor 3.76079 corresponde ao ponto y

Position = 4.15556 3.76079 1 0

ga-> q time → fornece o tempo fixado pelo usuário

Time = 00Z01JAN1979 to 00Z01JAN1979 Mon to Mon

clear ou **c** → limpa a tela do GrADS mantendo todas as configurações

reset → limpa a tela do GrADS sem fechar os arquivos

reinit → limpa a tela do GrADS e fecha todos os arquivos

quit → sai do GrADS

5. Iniciando o GrADS

Ao abrir um terminal do Linux, digite o comando “gradsnc” ou simplesmente “gradsc”. Uma nova janela deverá surgir e com isso você já está pronto para usar o GrADS e criar seus gráficos.

6. Abrindo arquivos

Existem alguns formatos de arquivo mais comuns usados no GrADSs, são eles: *.bin*, *.gra*, *.grb* e o *.nc*.

Para abrir arquivos “.ctl” usamos o comando **open** – **Exemplo** : ga-> open 9503.ctl

Para abrir arquivos “.nc” usamos o comando **sdfopen** – **Exemplo** : ga-> sdfopen 9503.nc

7. Visualizando arquivos

Alguns comandos precisam ser usados para visualizar informações sobre os arquivos abertos, são eles:

set – serve para especificar quando, onde e como os dados devem ser mostrados (**ga> set lat -20 10; set lon -100 -20; set t 1 20; set z 3, set lev 850, set time 00z22mar1998, set font 2 etc.**).

d ou **display**– mostra o campo desejado (**ga> d temp** , mostra o campo de temperatura, temp).

Importante: No GrADS, quando se abrem dois ou mais arquivos, cada um deles recebe uma numeração própria.

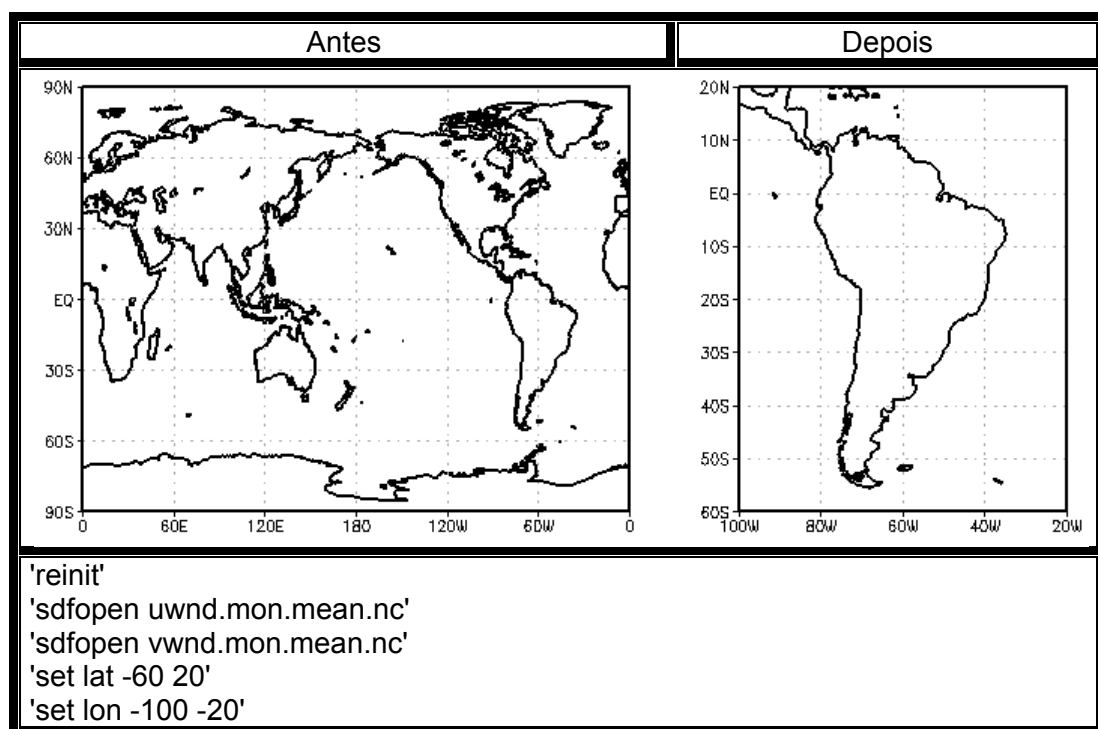
Exemplo:

```
'reinit'
'sdfopen uwnd.mon.mean.nc'      ;*abrindo o arquivo 1
'sdfopen vwnd.mon.mean.nc'      ;*abrindo o arquivo 2
'd uwnd'                        ;*por padrão, o primeiro arquivo aberto não precisa colocar 1 ao lado da variável
'c'                             ;*limpa a tela para mostrar a segunda variável
'd vwnd.2'                      ;*no caso do segundo arquivo aberto (vwnd.mon.mean.nc), ao visualizá-lo, deverá
                                ;ser colocado “.2” ao lado da variável. O GrADS entende que o usuário deseja visualizar a variável
                                ;do arquivo 2.
```

Importante: O sinal “;” (ponto e vírgula) é interpretado pelo GrADS como sendo a próxima linha do script “.gs”. O símbolo “*” (asterisco) representa um comentário no script “.gs”.

Exemplo:`'reinit'``*abrindo os arquivos``'sdfopen uwnd.mon.mean.nc'``'sdfopen vwnd.mon.mean.nc'``*visualizando os arquivos``'d uwnd'``'c' ;*limpa a tela do GrADS para mostrar a próxima variável``'d vwnd.2'`**8. Variando ou Fixando a latitude e longitude**

As latitudes variam de sul (valores negativos) para norte (valores positivos). As longitudes variam de oeste (valores negativos) para leste (valores positivos).

**9. Variando o nível de pressão**`ga-> set z 1``LEV set to 1000 1000``ga-> set z 2``LEV set to 925 925``ga-> set z 3``LEV set to 850 850``ga-> set z 17``LEV set to 10 10`

Outra forma de fixar os níveis.

```
ga-> set lev 1000  
LEV set to 1000 1000
```

```
ga-> set lev 925  
LEV set to 925 925
```

```
ga-> set lev 925 500  
LEV set to 925 500
```

10. Fixando ou variando o tempo

```
ga-> set t 1  
Time values set: 1979:1:1:0 1979:1:1:0  
ga-> set t 1 20  
Time values set: 1979:1:1:0 1980:8:1:0
```

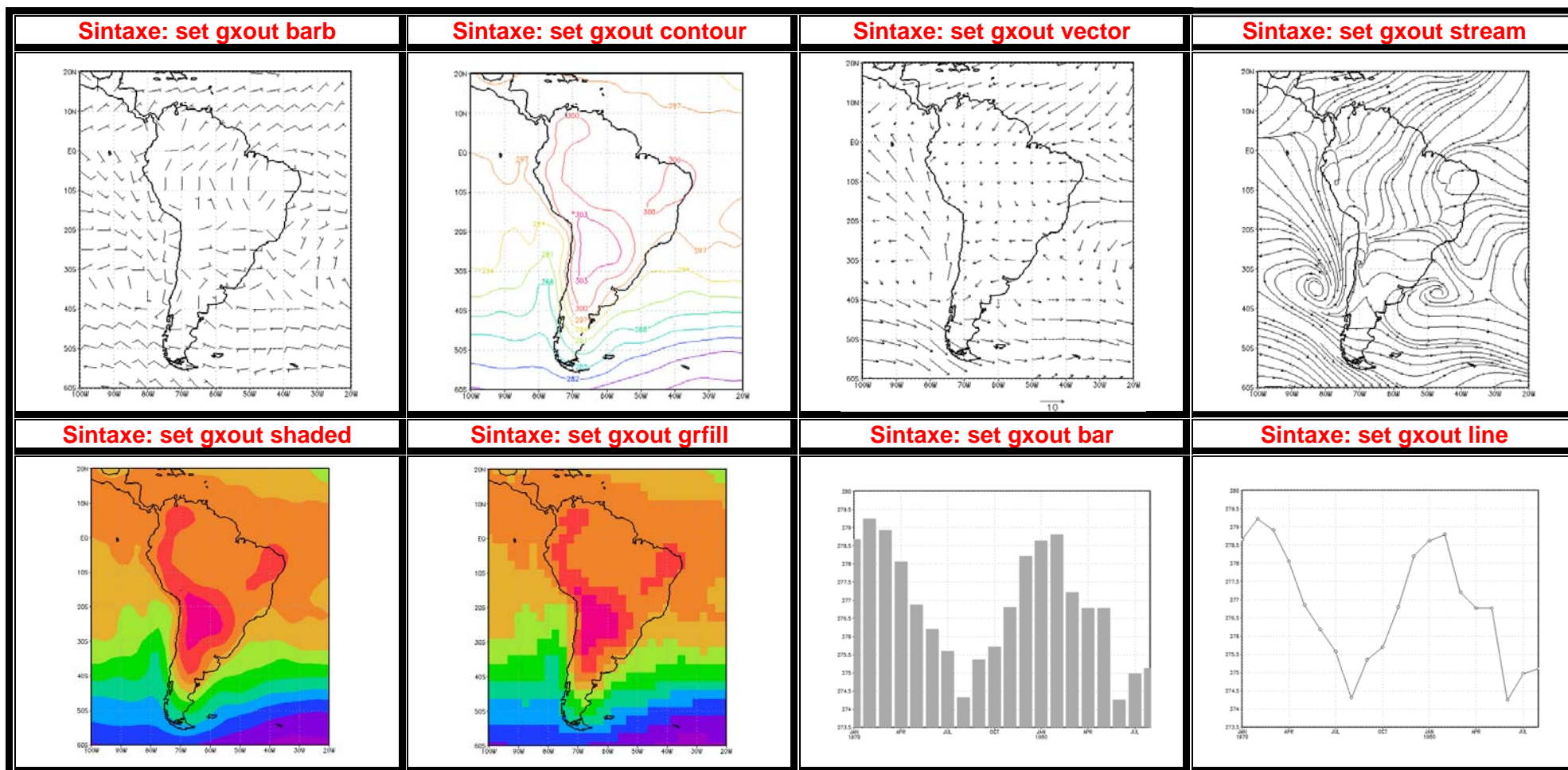
ou

```
ga->set time 00z01jan1950 00z20mar1990  
Time values set: 1950:1:1:0 1990:3:20:0
```

```
ga-> set time 00z01jan1950  
Time values set: 1950:1:1:0 1950:1:1:0
```

11. Tipos de gráfico

Existem vários tipos de saídas gráficas. Aqui, são mostrados apenas alguns modelos.



12. Cores usadas no GrADS


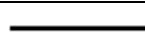
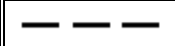

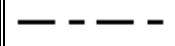
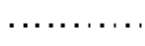
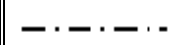
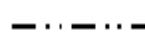
O GrADS disponibiliza 16 cores (tabela abaixo), porém você pode criar sua própria paleta de cores. Estas cores são usadas para modificar as cores das figuras como também marcadores, letras, linhas etc.

Exemplo: set ccolor 0 ou set ccolor 7

0		Black	1		white
2		Red	3		green
4		Blue	5		cyan
6		Magenta	7		yellow
8		Orange	9		purple
10		yellow/Green	11		med. blue
12		dark yellow	13		aqua
14		dark purple	15		grey

13. Estilos de linhas disponíveis no GrADS

Exemplo: set cstyle 4 ou set cstyle 7

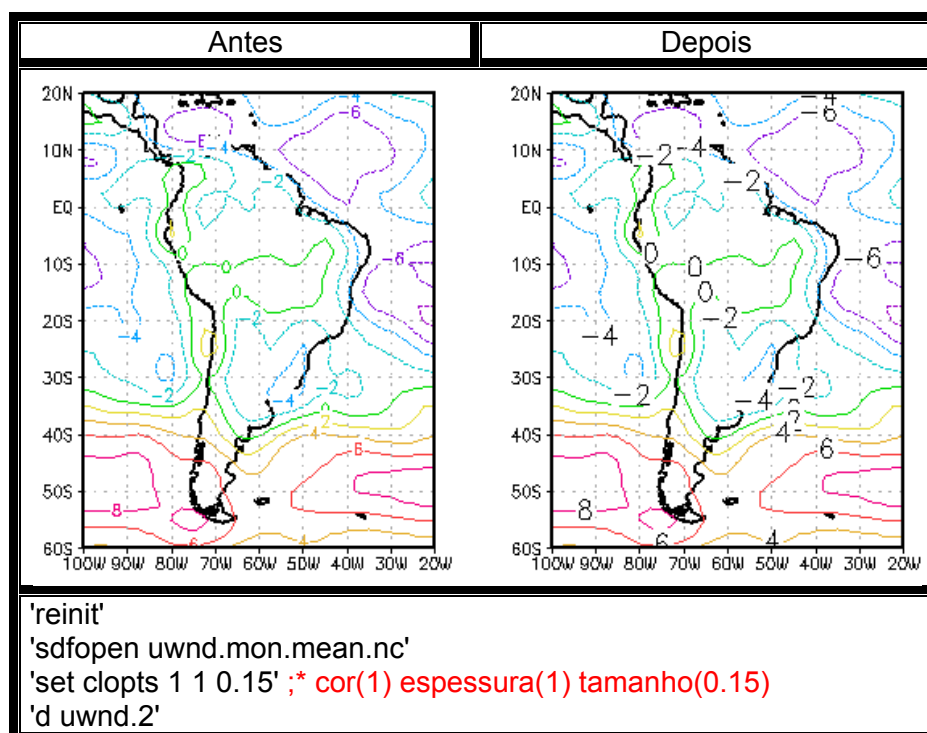
0		None	1		solid
2		long dash	3		short dash
4		long dash short	5		dots
6		dot dash	7		dot dot dash

14. Marcadores disponíveis no GrADS

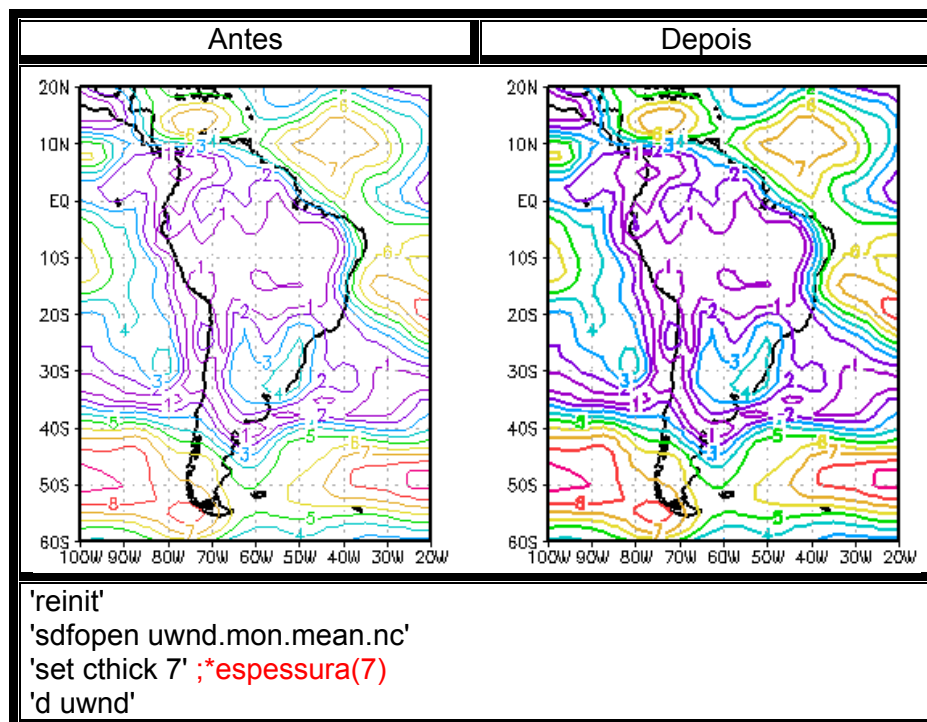
Exemplo: set cmark 6 ou set cmark 9

0		Nada	1	+	cruz
2	○	círculo aberto	3	●	círculo fechado
4	□	quadrado aberto	5	■	quadrado fechado
6	×	X	7	◇	diamante
8	△	triângulo aberto	9	▲	triângulo fechado
10	◐	círculo aberto com linha preta	11	◑	círculo aberto com linha branca

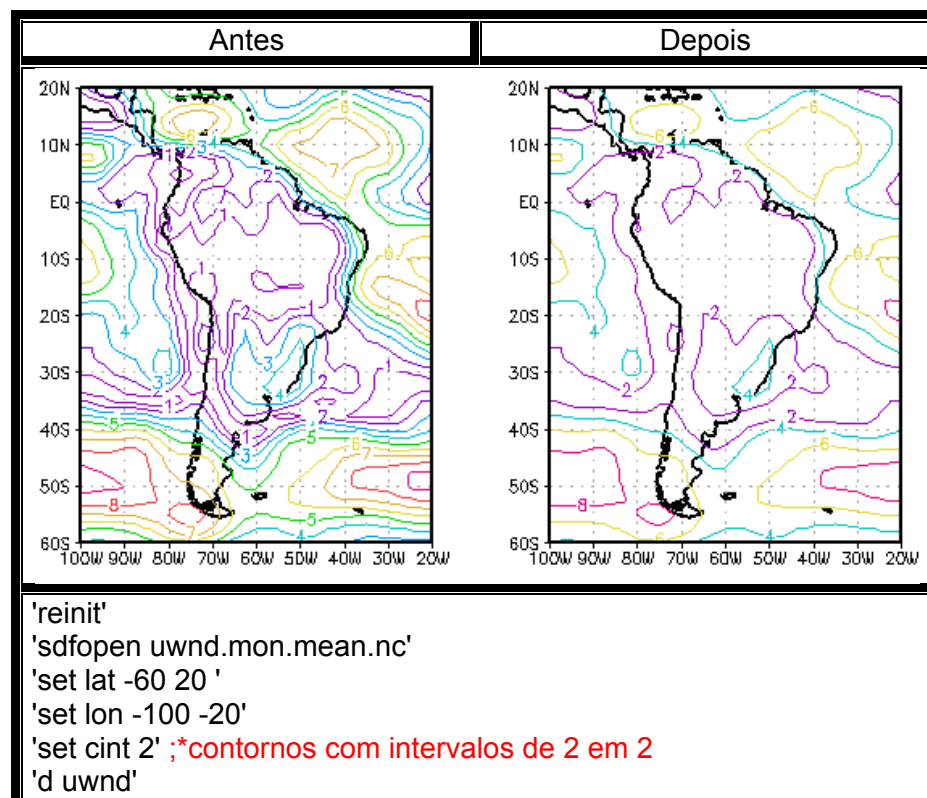
15. Modificando cor, espessura e tamanho dos rótulos dos contornos



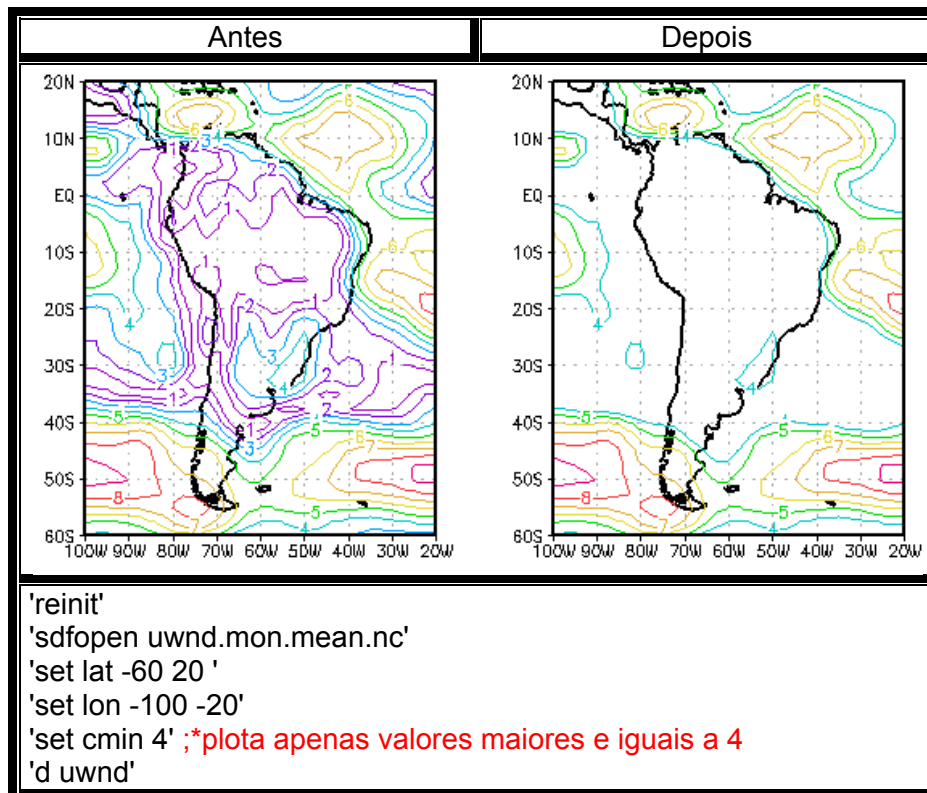
16. Modificando a espessura da linha do contorno



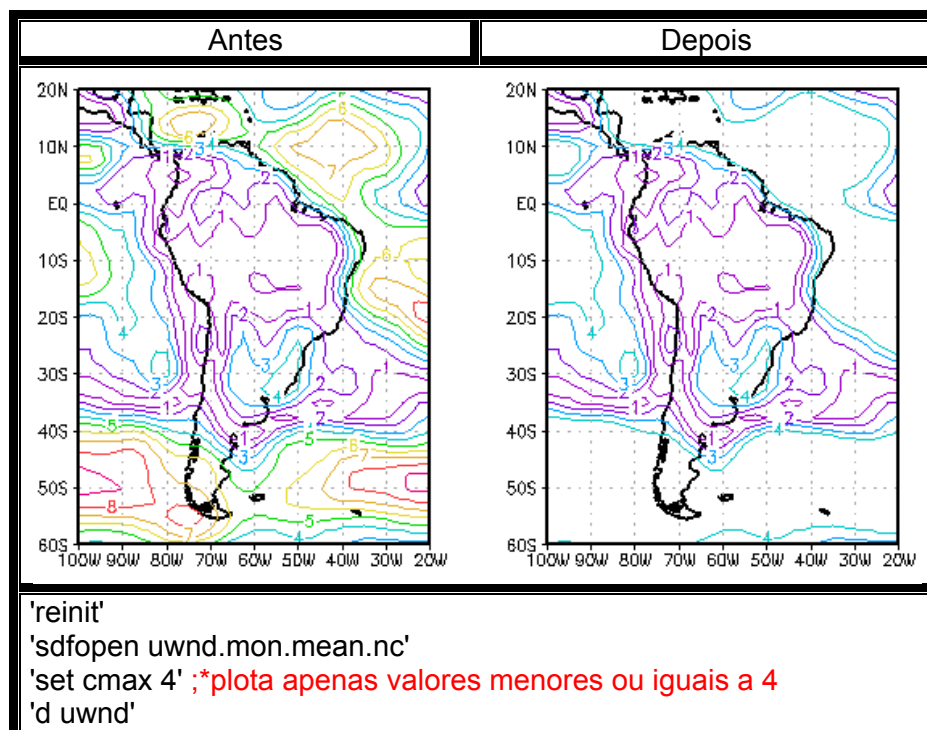
17. Modificando o intervalo dos contornos



18. Visualizando valores acima de um determinado valor



19. Visualizando valores abaixo de um determinado valor



20. Visualizando valores dado um intervalo específico

Sintaxe : set clevs val1 val2 val3 ... valn

onde: val1, val2, val3 são os valores limites que serão plotados.

Exemplo: Tem-se um campo de vento zonal e deseja-se plotar apenas os valores de 1 até 6 m.s⁻¹. Para isso, basta fazer : 'set clevs 1 2 3 4 5 6'. Este procedimento faz com que sejam plotados apenas os valores de 1 a 6.

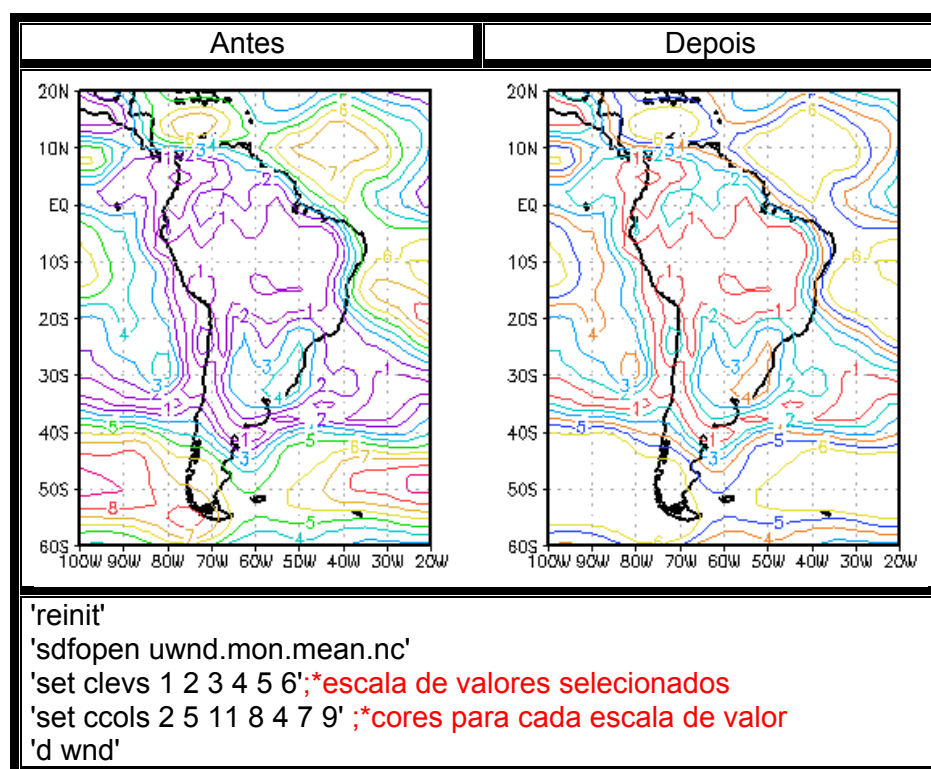
Atribuindo cores aos valores especificados. Isto é útil para definir intervalos e atribuir cores aos valores específicos.

Sintaxe: set ccols val1 val2 val3 ... valn

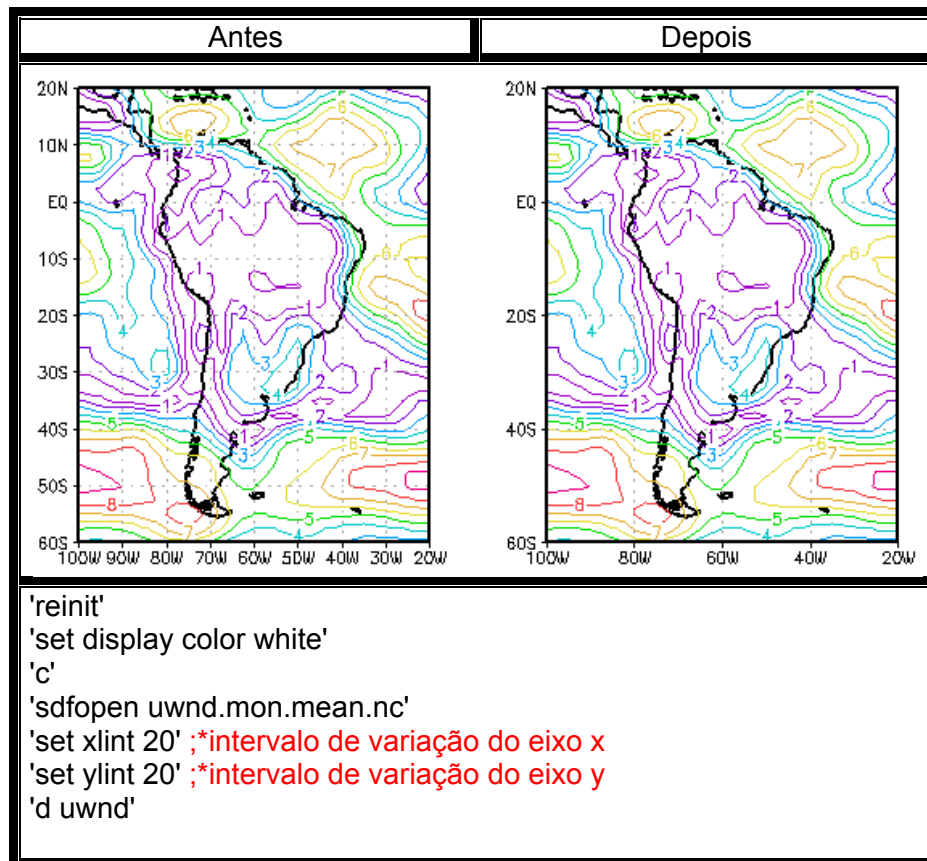
onde: val1, val2, val3 = número das cores (ver tabela de cores) correspondente a cada nível

Exemplo: Têm-se os intervalos de valores (seja vento, temperatura etc.) variando de 1 até 6, totalizando 6 intervalos, logo teremos que selecionar 7 cores.

Interpretação: Uma cor será para valores abaixo de 1, uma cor entre 1 e 2, uma cor entre 2 e 3, uma cor entre 3 e 4, uma cor entre 4 e 5, uma cor entre 5 e 6 e uma cor para valores acima de 6. Totalizando assim 7 cores. As cores são selecionadas pelo usuário.

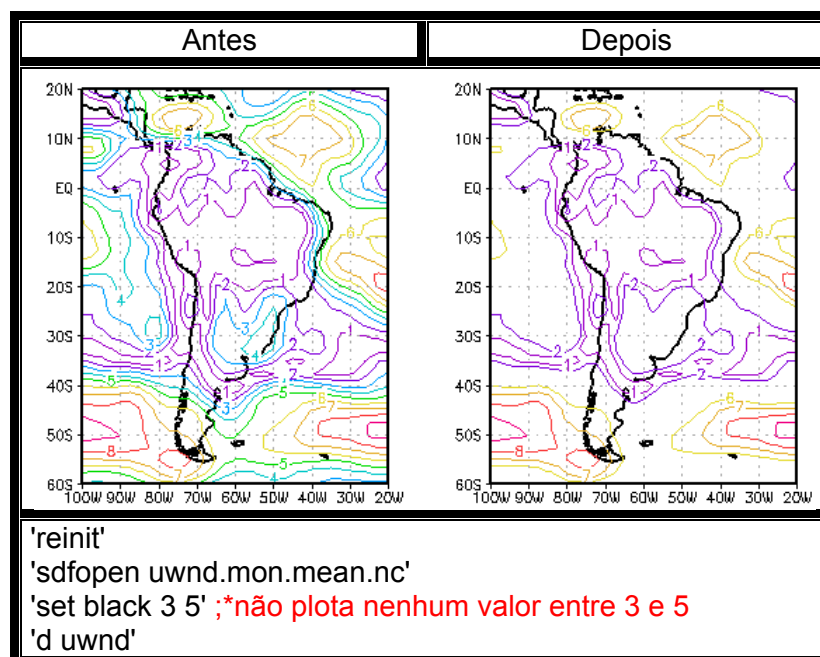


21. Especificando o intervalo de variação do eixo x e y



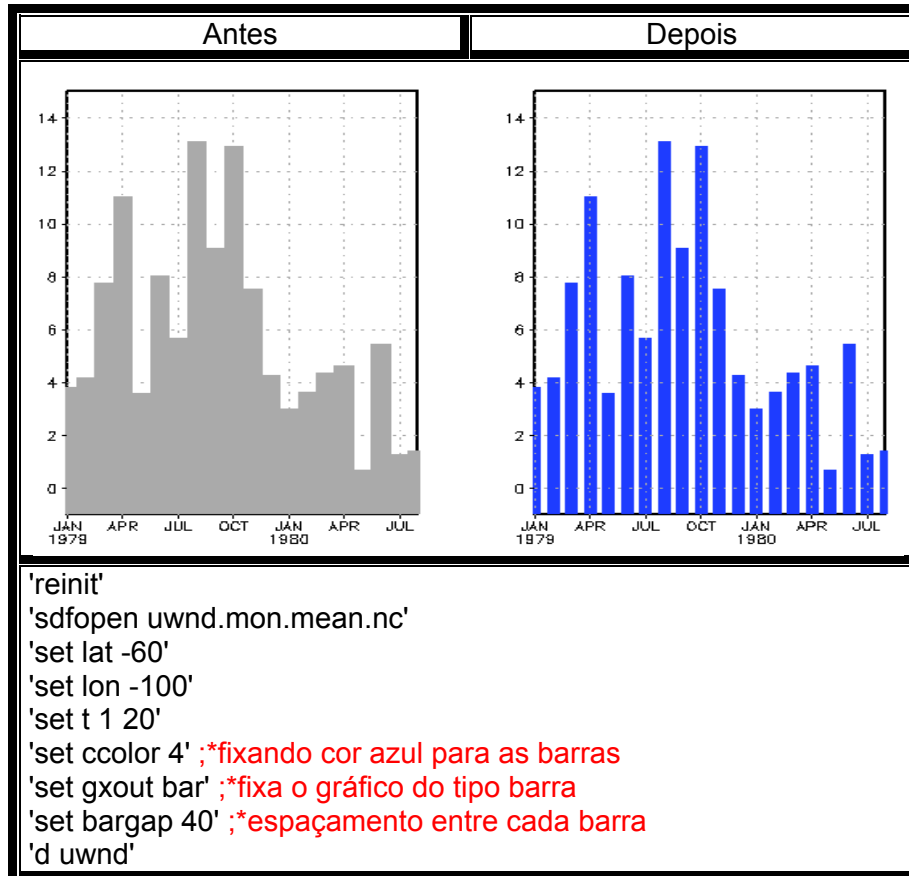
22. Visualizando valores fora de um limite especificado.

Por exemplo, desejamos visualizar os valores abaixo de 3 e acima de 5. Vejamos o exemplo abaixo para fixação do comando.

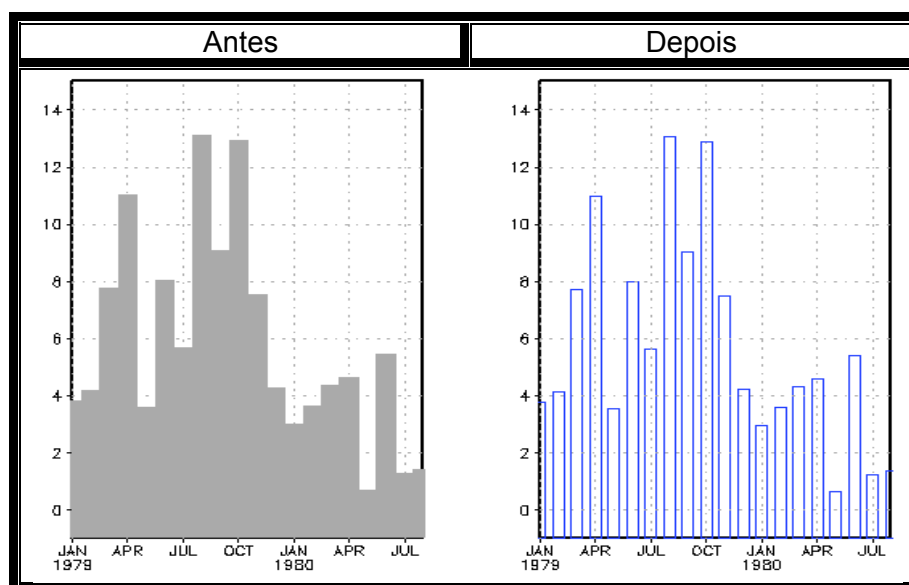


23. Gráfico de Barra

Atribuindo espaço entre as barras (set bargap val). Onde: val = valor em porcentagem de 0-100. Por padrão, o gráfico de barras é preenchido. Para o caso de barras sem preenchimento, usar a opção para não preenchimento (outline).



24. Gráfico de barras sem preenchimento

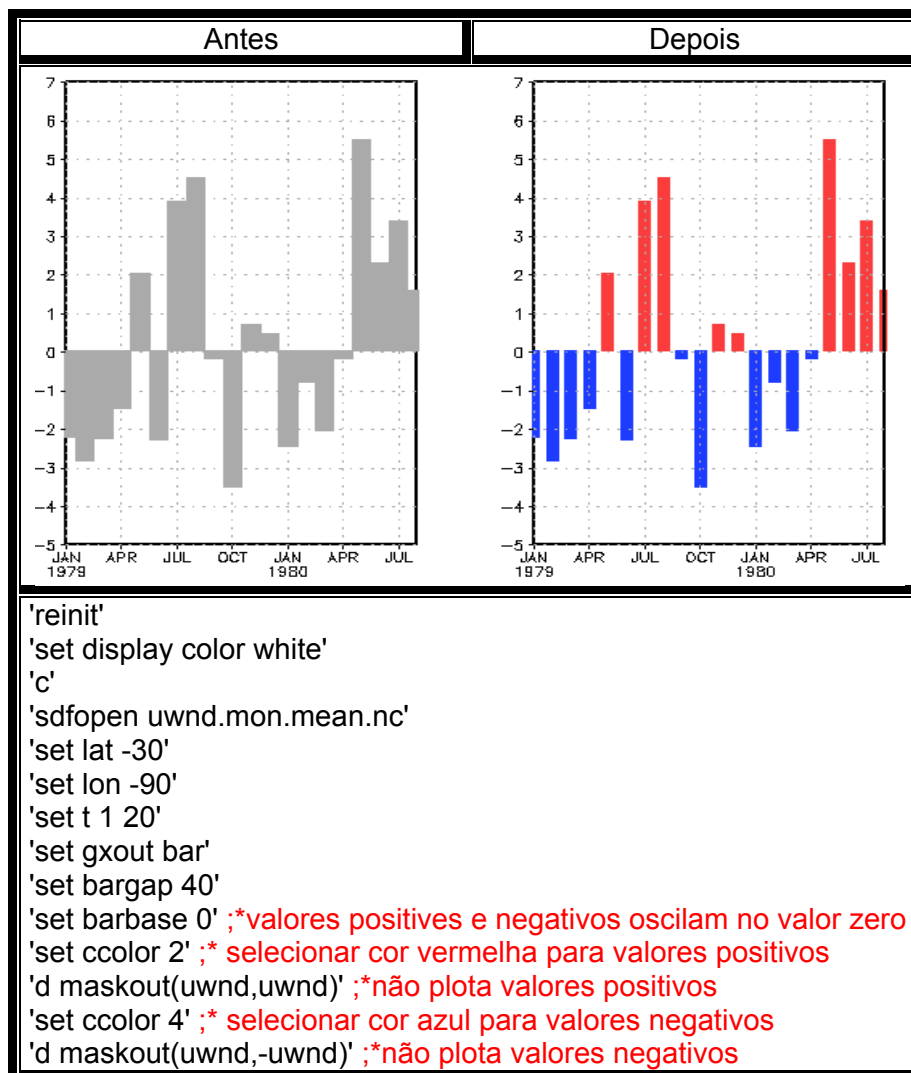


```
'reinit'
'sdfopen uwnd.mon.mean.nc'
'set lat -60'
'set lon -100'
'set t 1 20'
'set ccolor 4'
'set gxout bar'
'set bargap 40'
'set baropts outline' ;*barras sem preenchimento
'd uwnd'
```

Importante: O comando “barbase” é usado em gráficos de barras, o qual é útil quando se deseja fazer gráficos que apresentam valores positivos e negativos.

Sintaxe: set barbase val. Onde: val = valor desejado. Caso deseja-se fazer um gráfico com valores positivos e negativos é aconselhável usar o valor 0.

Exemplo: set barbase 0

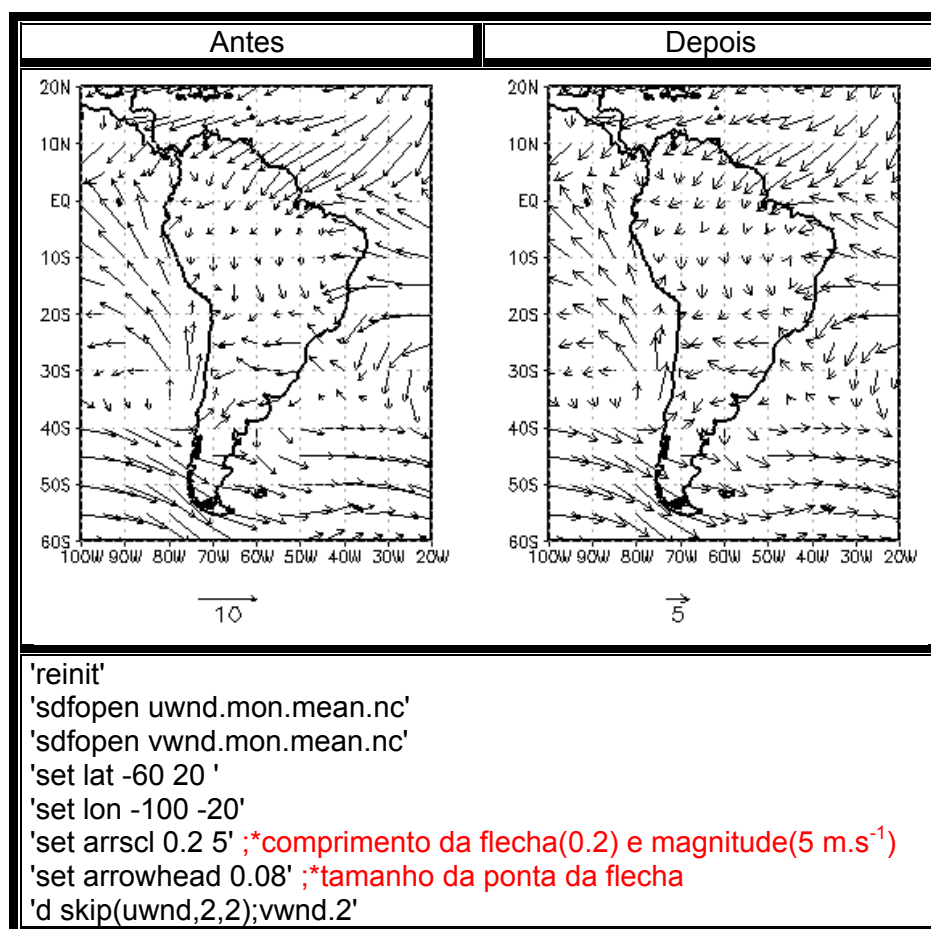


25. Gráfico de flecha do vento

25.1. Comprimento da escala da flecha

Sintaxe: set arrscl tamanho <magnitude>

O tamanho, é o comprimento da flecha em unidades de plotagem (polegadas na página virtual) e deve estar entre 0.5 e 1.0. A magnitude é a grandeza do vetor que produzirá uma flecha de especificado tamanho. Os outros comprimentos de flecha serão escalados apropriadamente. Se a magnitude não é dada, todas terão o mesmo comprimento.



26. Escrevendo texto no gráfico

26.1. Adicionando título

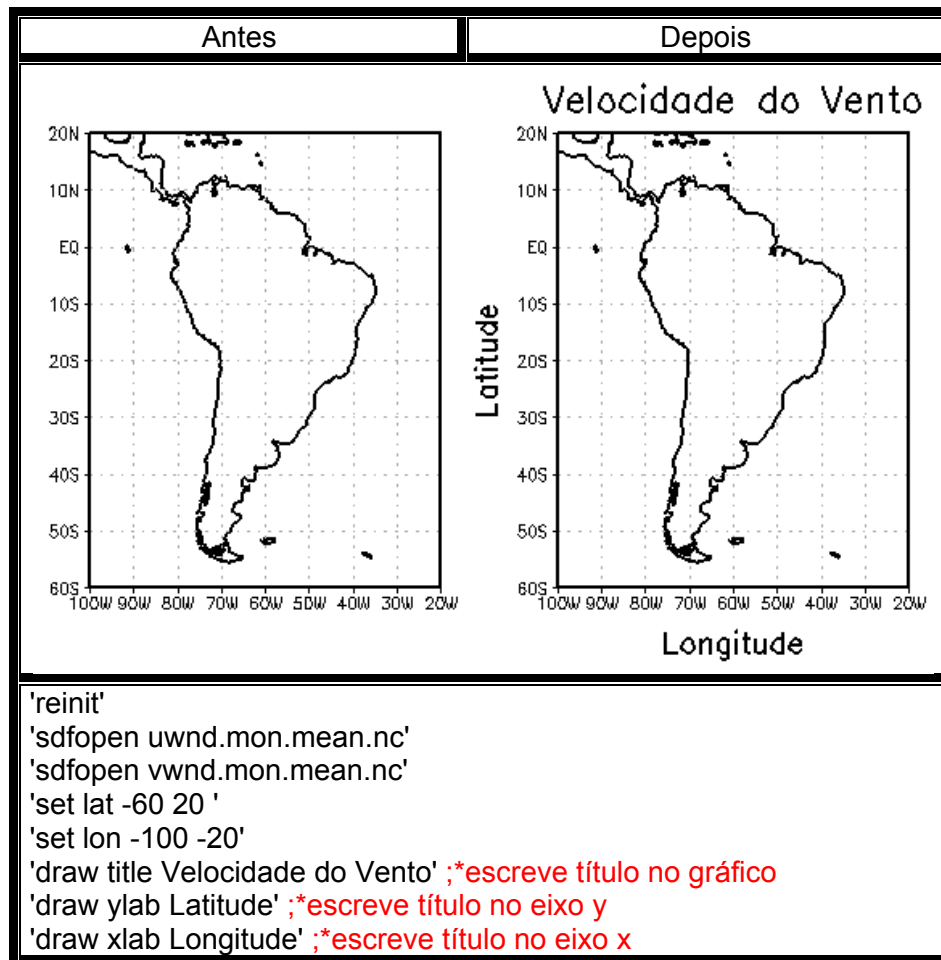
Sintaxe: draw title text

26.2. Adicionando título aos eixos x e y

Sintaxe: draw xlab text

Sintaxe: draw ylab texto

Onde “texto” é o que será escrito no gráfico.



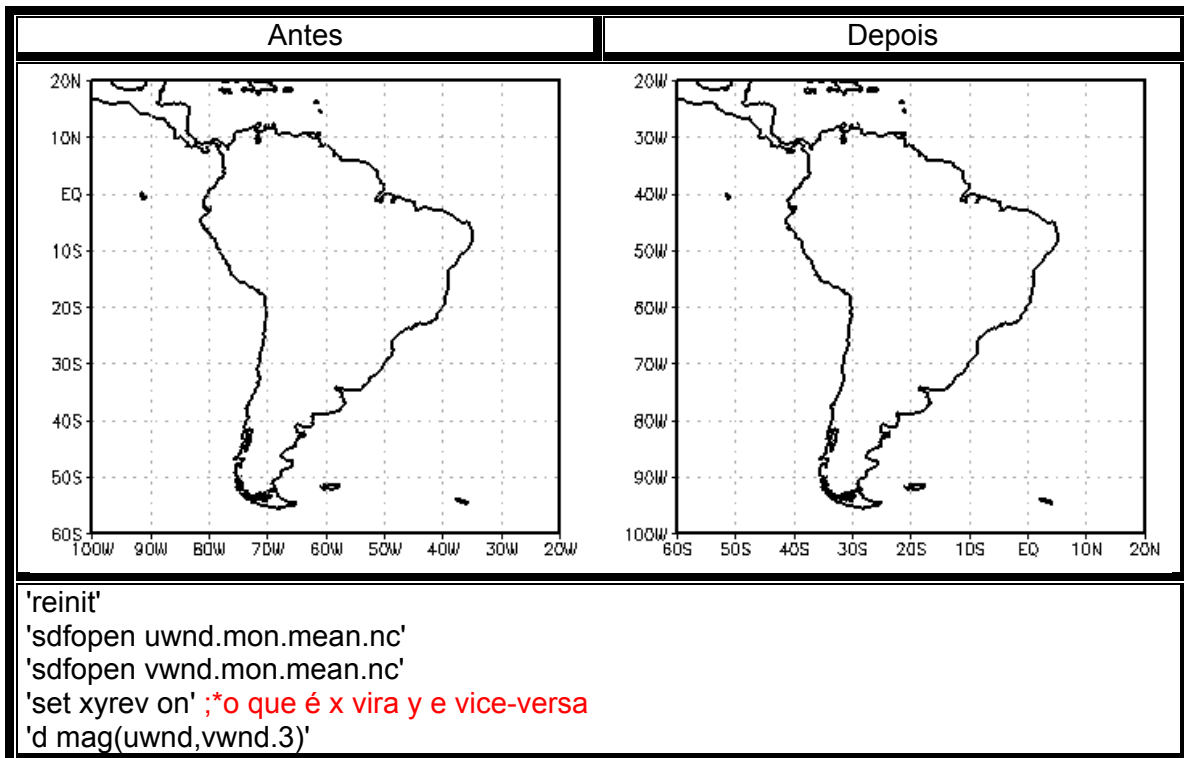
27. Invertendo os eixos x e y em um gráfico

O que é x vira y e vice-versa. Apenas os eixos são afetados e nada acontece com o gráfico.

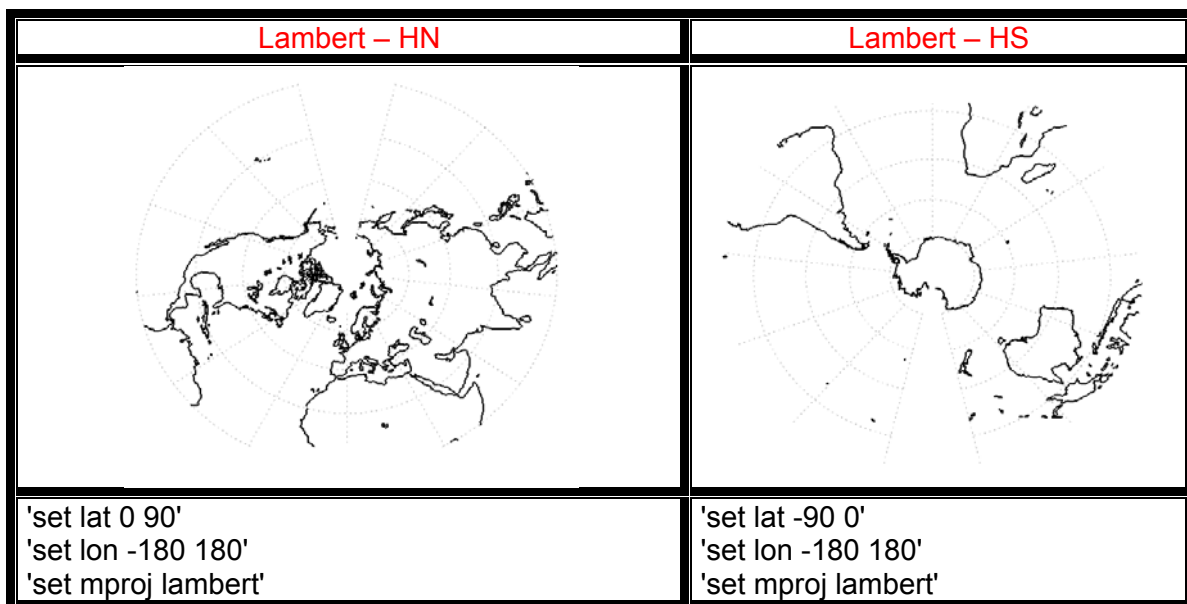
Sintaxe: set xyrev on – ativa

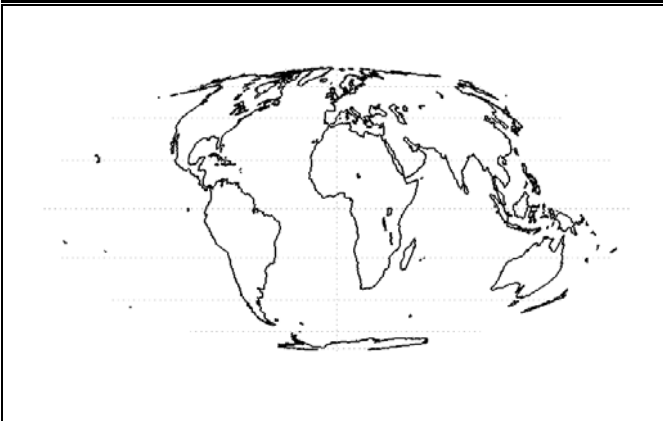
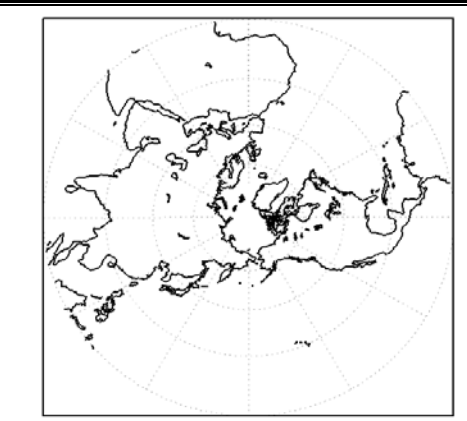
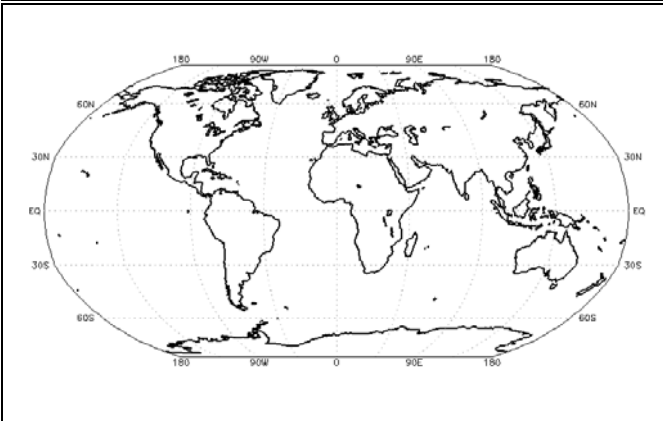
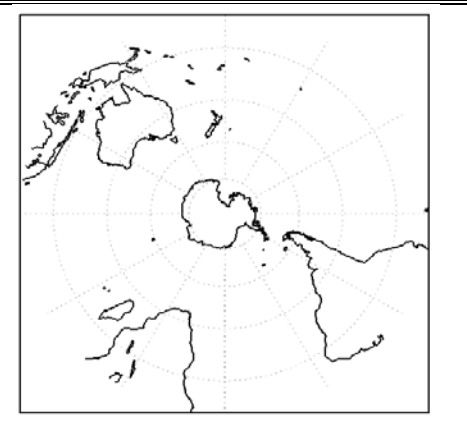
e

Sintaxe: set xyrev off – desativa



28. Projeções disponíveis no GrADS



Mollweide	Estereográfica Polar – HN
	
'set lat -90 90' 'set lon -180 180' 'set mproj mollweide'	'set lat 0 90' 'set mproj nps'
Robinson	Estereográfica Polar – HS
	
'set lat -90 90' 'set lon -180 180' 'set mproj robinson'	'set lat -90 0' 'set mproj sps'

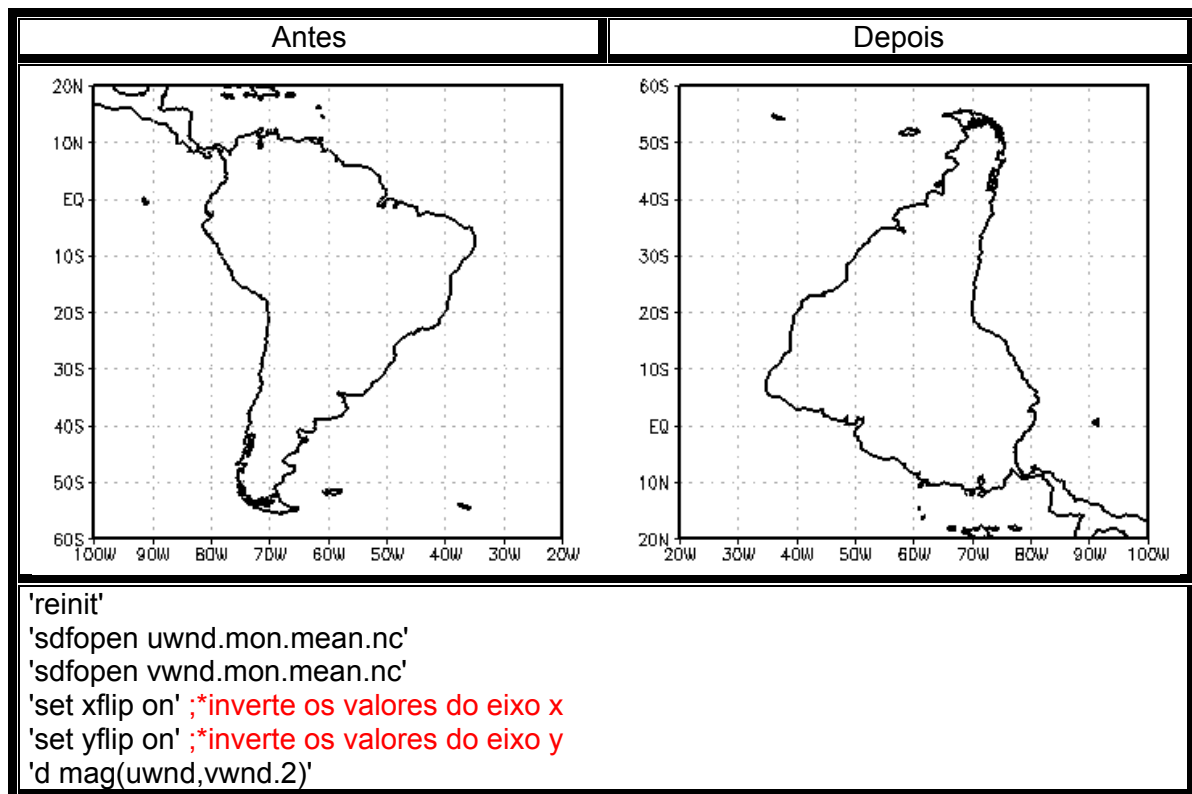
29. Invertendo a ordem dos eixos x e/ou y

Diferentemente do comando “set xyrev” os eixos e os mapas são modificados.

Para o eixo x: **Sintaxe:** set xflip on ou set xflip off

Para o eixo y: **Sintaxe:** set yflip on ou set yflip off

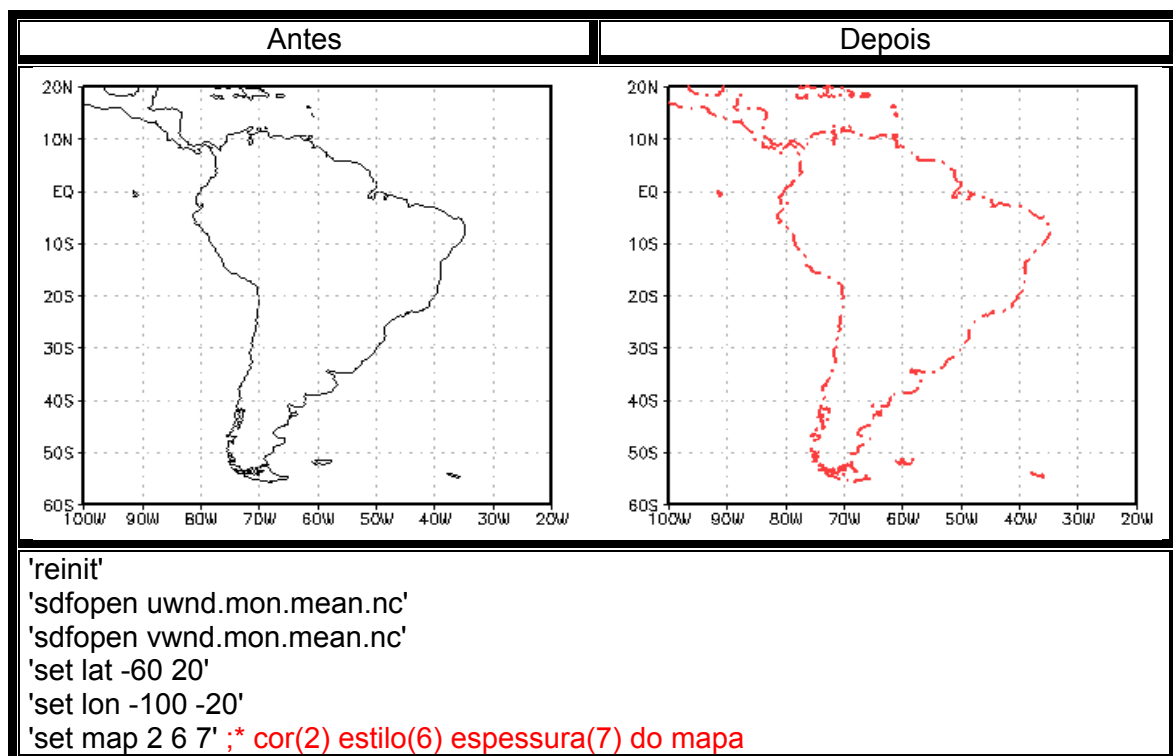
Exemplo: Têm-se os valores 1 2 3 4 5 6 7 para o eixo x ou y. Ao ativar este comando estes valores passam a ter a seguinte disposição 7 6 5 4 3 2 1.



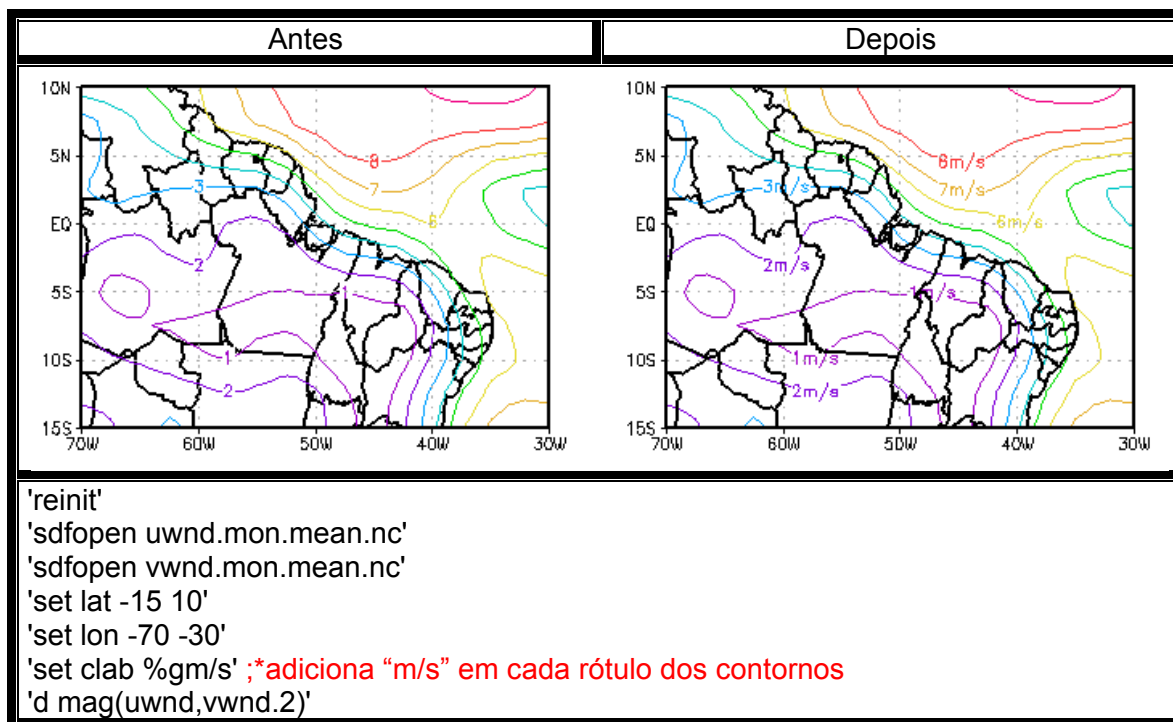
30. Atribuindo características ao mapa

Este comando atribui “cor”, “estilo” e “espessura” ao contorno do mapa.

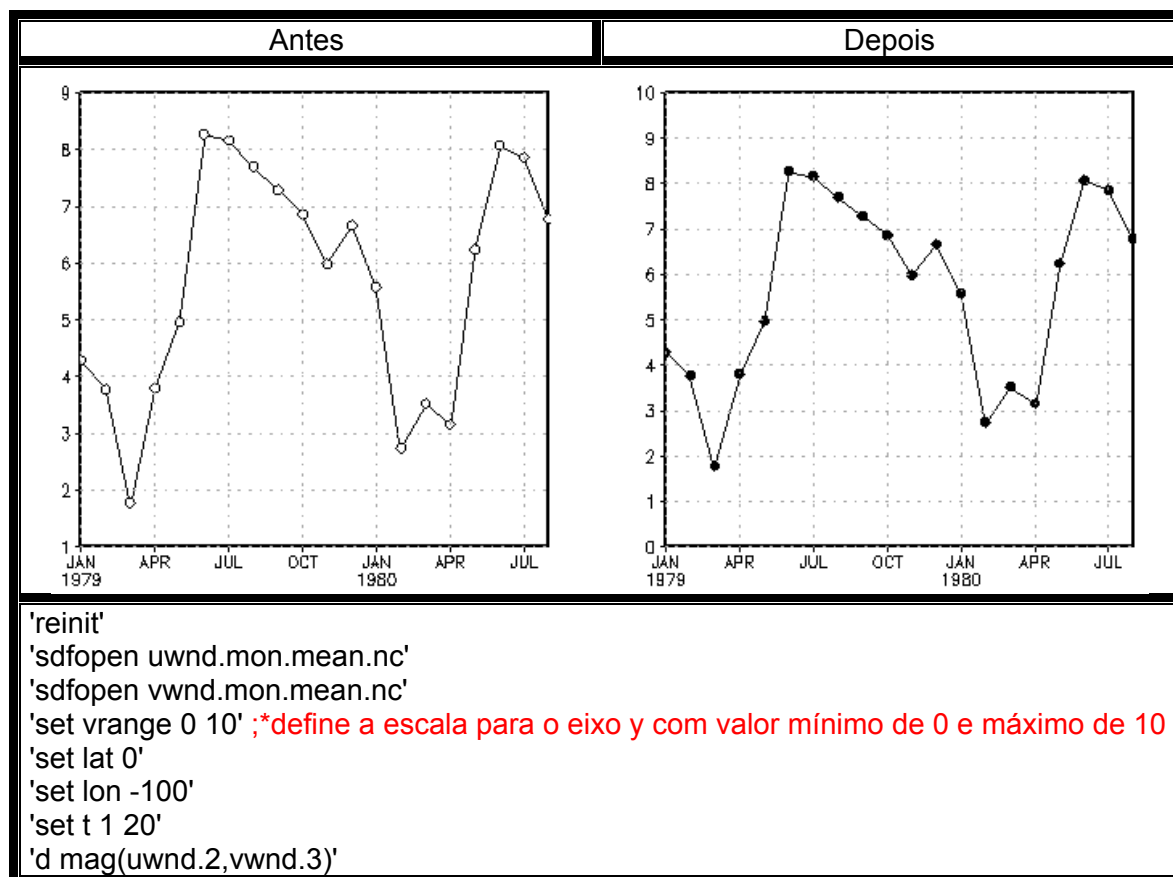
Sintaxe: set map cor estilo espessura



31. Adicionando strings nos rótulos dos contornos

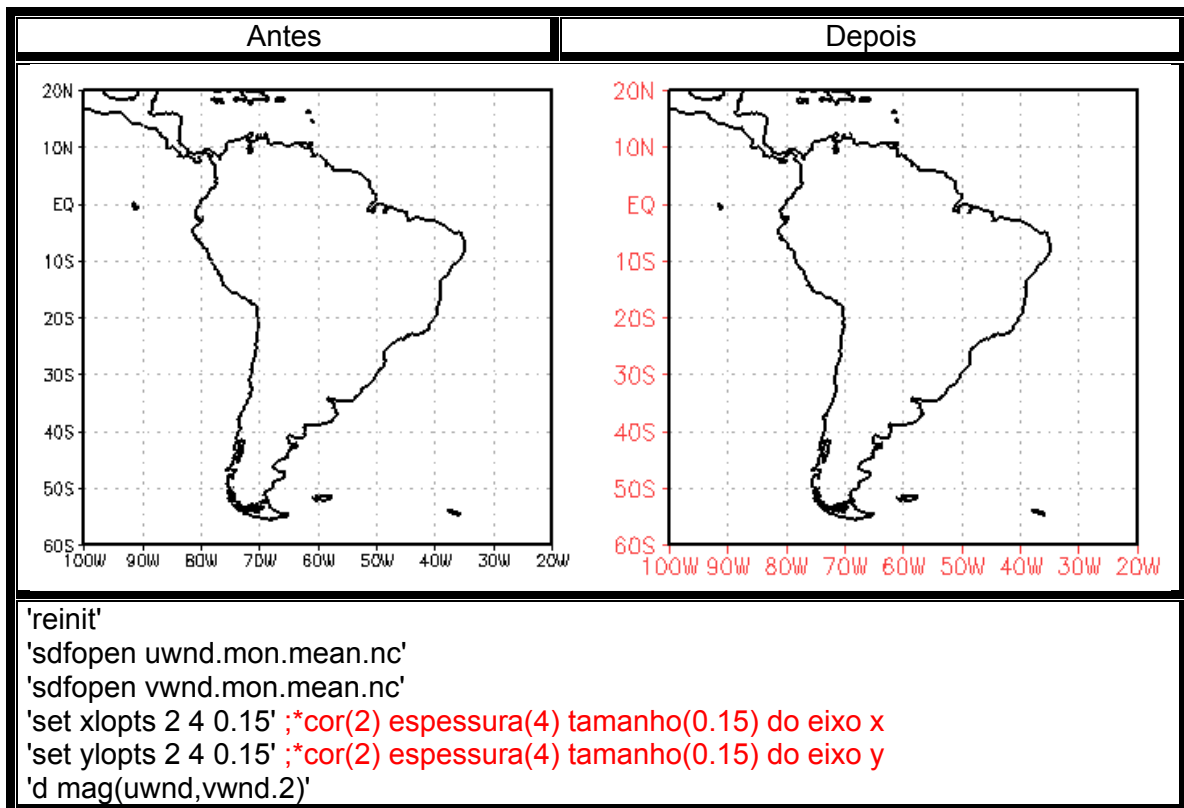


32. Fixando valores para o eixo y



Importante: O comando anterior manipula o eixo y. Caso a mudança fosse aplicada ao eixo x, deve-se usar o comando `'set vrange2'`.

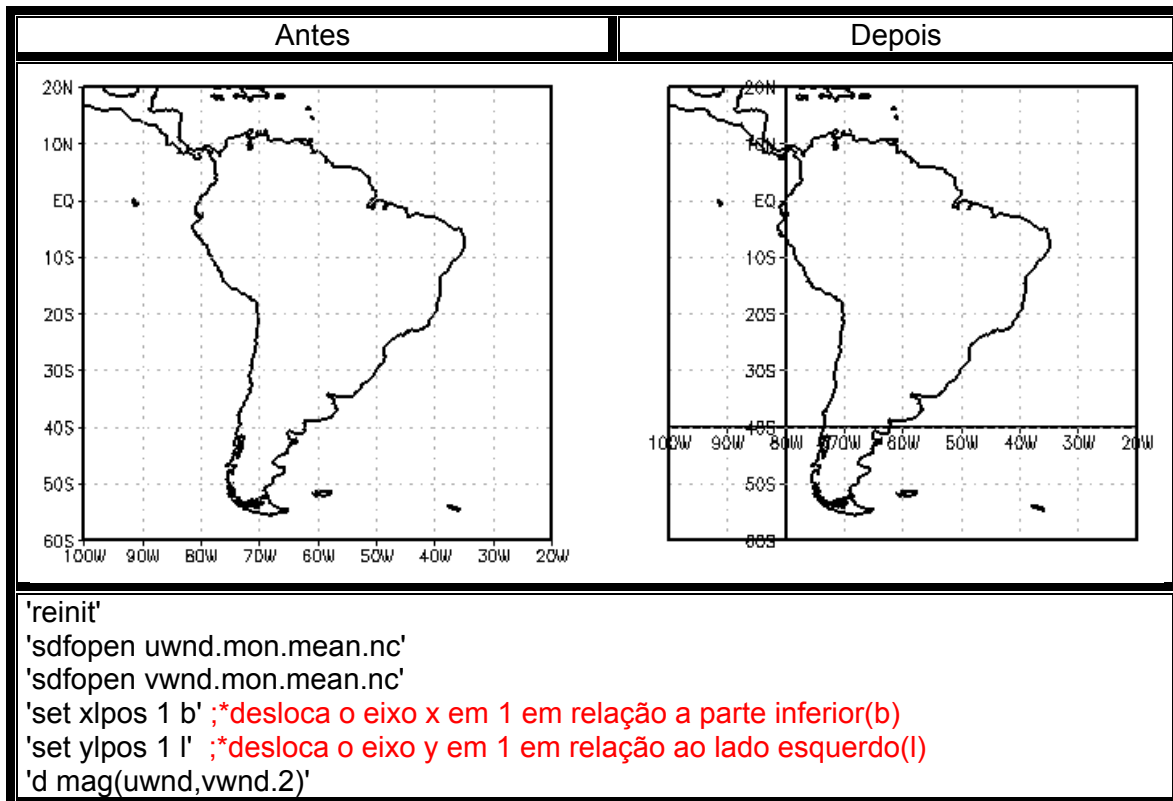
33. Alterando características dos eixos x e y



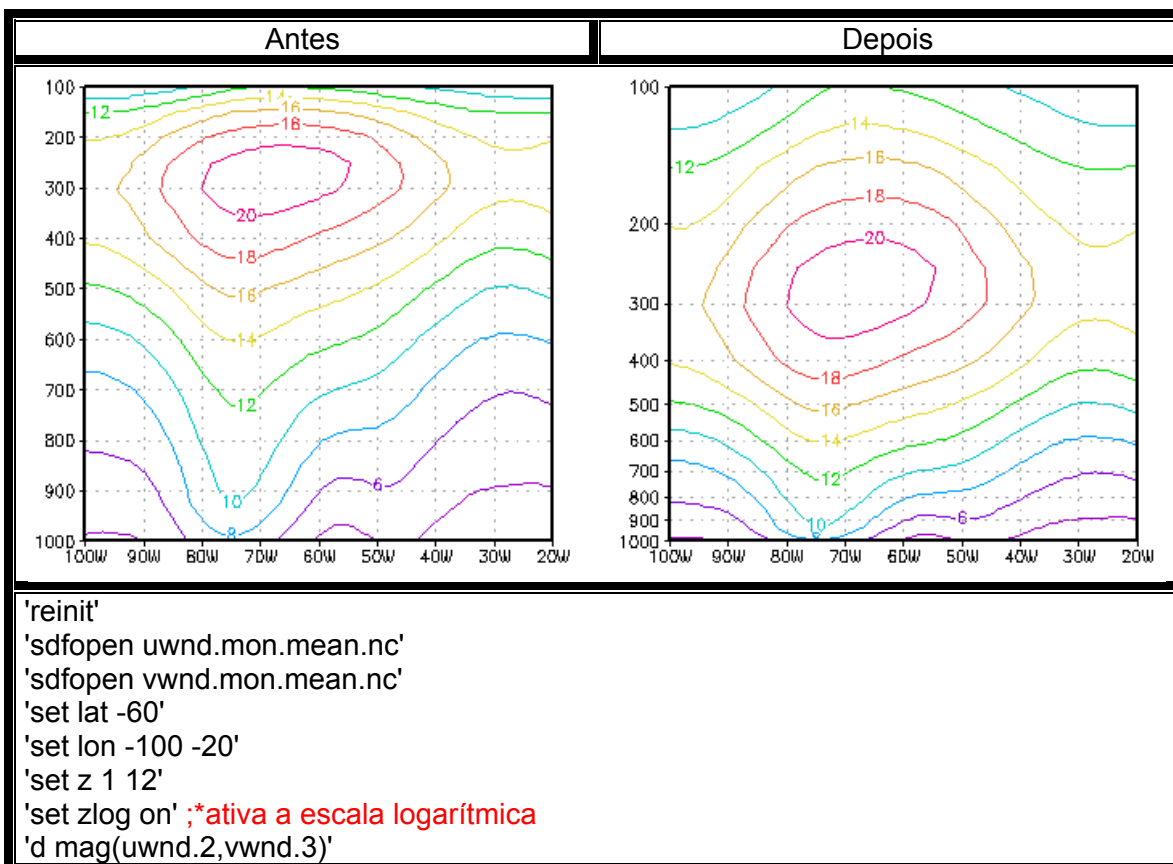
34. Deslocando o eixo x e/ou y em um gráfico

Para o eixo x: **Sintaxe:** `set xlpes val1 val2`. Onde val1 = corresponde ao valor que o eixo deve ser deslocado em relação a val2 que pode ser **b** (bottom = lado inferior) ou **t** (top = lado superior).

Para o eixo y: **Sintaxe:** `set ylpes val1 val2`. Onde val1 = corresponde ao valor que o eixo deve ser deslocado em relação a val2 que pode ser **l** (left = lado esquerdo) ou **r** (right = lado direito).



35. Gerando um gráfico de corte vertical (Hovmoller) com escala logarítmica



36. Obtendo as coordenadas x e y de um ponto

Dado um ponto (latitude e longitude) e desejamos saber os valores em ponto de grade deste ponto. Para tal, basta usar o comando “w2xy”. Ele converte pontos globais em coordenadas x e y da página.

Exemplo: Para desenhar um marcador (veja tabela de marcadores) em um ponto específico do gráfico será usado o comando “w2xy”.

Dados: **latitude = -10** (10°S) e **longitude = -50** (50°W)

No terminal do GrADS digitamos o comando abaixo. Lembrando que a sequência a ser digitada é longitude (x) e depois latitude (y).

ga->q w2xy -50 -10

Serão retornados os pontos correspondentes a “x” e “y” respectivamente.

37. Desenhando um marcador

Sintaxe: draw mark <tipo_marcaador> <X> <Y> <tamanho>

Onde: tipo de marcaador = ver tabela de marcadores

X e Y = coordenadas obtidos com o comando w2xy

tamanho = 0.1, 0.2, 0.3, ...

Exemplo:

ga->draw mark 5 3.34 6.67 0.15

Dessa forma, será criado um marcador do tipo quadrado fechado (5) na posição x = 3.34 e y = 6.67 com o tamanho 0.15. As coordenadas “x = 3.34 e y = 6.67” foram obtidas com o comando “q pos”.

38. Funções intrínsecas do GrADS

38.1. Capturando uma linha usando a função “sublin”

Sintaxe: sublin(string,linha)

Exemplo:

```
'sdfopen uwnd.mon.mean.nc'
'sdfopen vwnd.mon.mean.nc'
'set lat -60 20'
'set lon -100 -20'
'd mag(uwnd.2,vwnd.3)'
'q dims'
```

Ao digitar 'q dims' serão mostradas as informações abaixo:

```
ga-> q dims
Default file number is: 1           →linha 1
X is varying  Lon = -100 to -20  X = -39 to -7   →linha 2
Y is varying  Lat = -60 to 20   Y = 13 to 45     →linha 3
Z is fixed    Lev = 1000  Z = 1                →linha 4
T is fixed    Time = 00Z01JAN1979  T = 1         →linha 5
```

Capturando as linhas do comando “q dims”

```
linha2 = sublin(result,2) → X is varying  Lon = -100 to -20  X = -39 to -7
linha3 = sublin(result,3) → Y is varying  Lat = -60 to 20   Y = 13 to 45
linha4 = sublin(result,4) → Z is fixed    Lev = 1000  Z = 1
linha5 = sublin(result,5) → T is fixed    Time = 00Z01JAN1979  T = 1
```

38.2. Capturando uma palavra usando a função “subwrđ”

Sintaxe: subwrđ(string,palavra)

Onde: string = string a ser armazenada

palavra = é um valor inteiro que corresponde a posição da string a ser armazenada

Exemplo:

```
'sdfopen uwnd.mon.mean.nc'
'sdfopen vwnd.mon.mean.nc'
'set lat -60 20'
'set lon -100 -20'
'd mag(uwnd.2,vwnd.3)'
'q dims'
```

Ao digitar 'q dims' serão mostradas na janela as informações abaixo:

```
ga-> q dims
Default file number is: 1           →linha 1
X is varying  Lon = -100 to -20  X = -39 to -7   →linha 2
Y is varying  Lat = -60 to 20   Y = 13 to 45     →linha 3
Z is fixed    Lev = 1000  Z = 1                →linha 4
T is fixed    Time = 00Z01JAN1979  T = 1         →linha 5
```

Capturando a linha4 do comando 'q dims'

linha4 = sublin(result,4) → ^{1 2 3} Z is fixed ^{4 5 6} Lev = 1000 ^{7 8 9} Z = 1 → posições que serão usadas

Capturando parte de uma linha. Neste caso, a linha capturada foi a linha4.

p4 = subwrd(linha4,6) → observe que p4 armazenou da linha4 na posição 6 que corresponde a 1000. A variável "p4" poderia receber qualquer outro nome.

zz = subwrd(linha4,9) → veja que agora o valor armazenada da linha4 esta na posição 9 que corresponde a 1.

38.3. Capturando parte de uma palavra usando a função "substr"

Sintaxe: substr(string,início,tamanho)

Onde: string = string a ser armazenada

início = valor inteiro que indica a posição onde começa a palavra a ser armazenada

tamanho = valor inteiro que indica quantos caracteres terá a string a ser armazenada

Exemplo:

```
'sdfopen uwnd.mon.mean.nc'
'sdfopen vwnd.mon.mean.nc'
'set lat -60 20'
'set lon -100 -20'
'd mag(uwnd.2,vwnd.3)'
'q dims'
```

Ao digitar 'q dims' serão mostradas na janela as informações abaixo:

```
ga-> q dims
Default file number is: 1          →linha 1
X is varying  Lon = -100 to -20  X = -39 to -7      →linha 2
Y is varying  Lat = -60 to 20   Y = 13 to 45        →linha 3
Z is fixed    Lev = 1000  Z = 1          →linha 4
T is fixed    Time = 00Z01JAN1979  T = 1          →linha 5
```

Capturando a linha5 do comando 'q dims'

linha5 = sublin(result,5) → ^{1 2 3} T is fixed ^{4 5} Time = 00Z01JAN1979 ⁶ T = 1 ^{7 8 9} → posições

Capturando parte de uma linha. Neste caso, a linha capturada foi a linha5.

x1 = subwrd(linha5,6) → observe que x1 armazenou da linha5 na posição 6 que corresponde a "00Z01JAN1979".

Serão acrescentados espaços em "00z01JAN1979" para melhor visualização da função "substr"

^{1 2 3 4 5 6 7 8 9 10 11 12} → posições
0 0 Z 0 1 J A N 1 9 7 9

hora = substr(x1,1,2) → isso armazenara o valor "00" pois ele começa na posição um e têm dois caracteres (00). Lembre-se, no GrADS a hora apresenta apenas dois caracteres.

dia = substr(x1,4,2) → isso armazenara o valor “01” pois ele começa na posição quatro e têm dois caracteres (01). Lembre-se, no GrADS o dia tem no máximo dois caracteres.

mes = substr(x1,6,3) → isso armazenara o valor “JAN” pois ele começa na posição seis e têm três caracteres (JAN). Lembre-se, no GrADS os meses apresentam 3 caracteres. O idioma dos meses são dados em inglês.

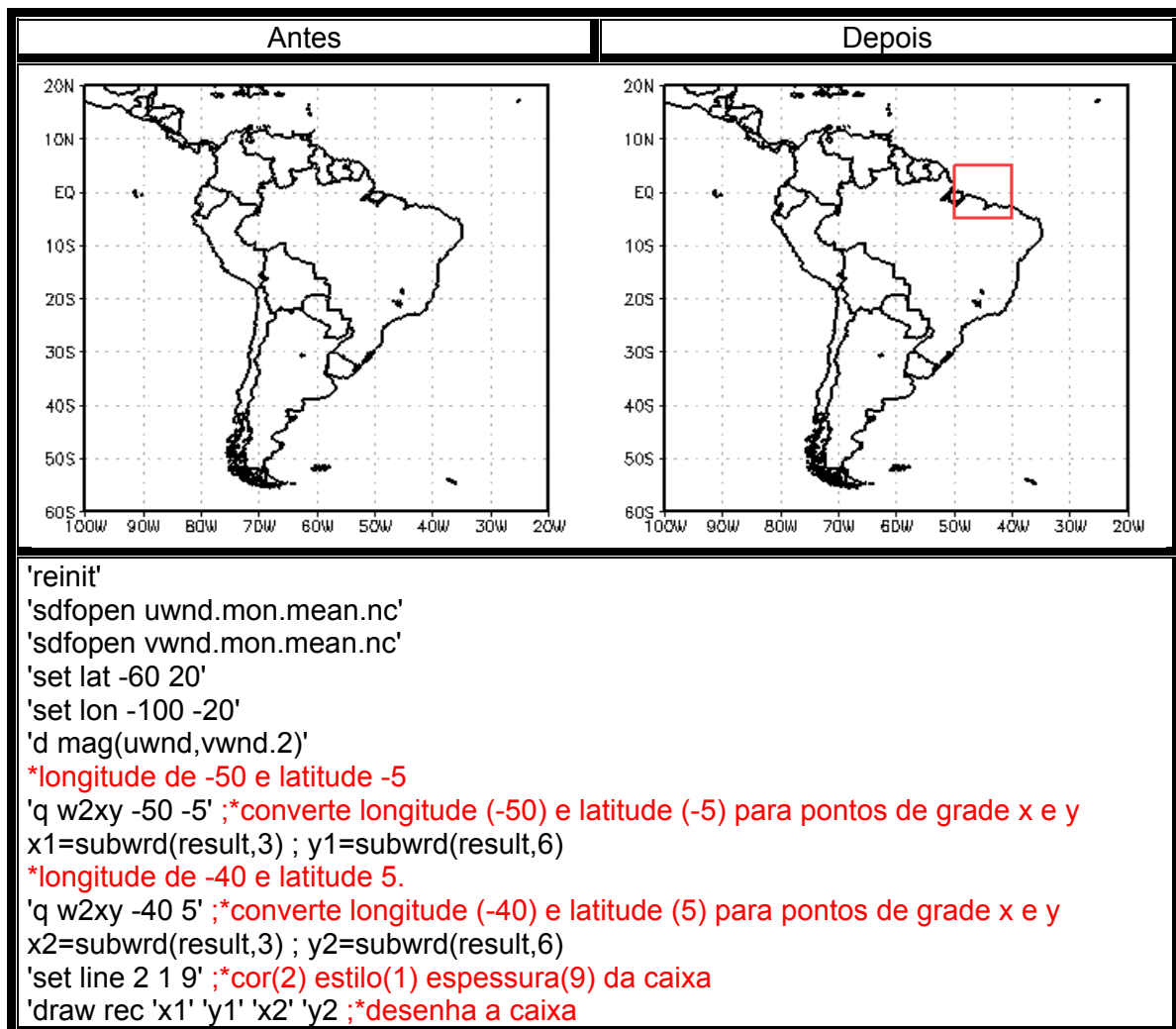
ano = substr(x1,9,4) → isso armazenara o valor “1979” pois ele começa na posição nove e têm quatro caracteres (1979). Lembre-se, no GrADS o ano apresenta 4 caracteres.

Conclusão:

Estas funções servem para automatizar os scripts armazenando valores, variáveis etc. Isso dependerá das necessidades do usuário.

39. Desenhando uma caixa no gráfico

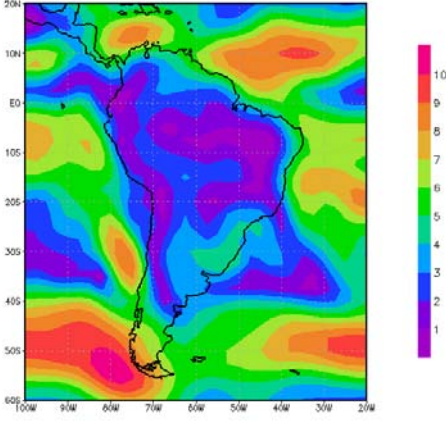
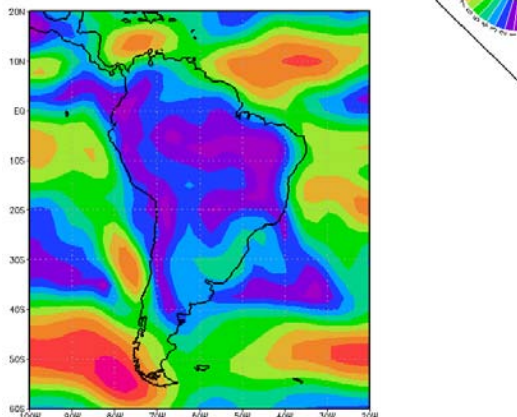
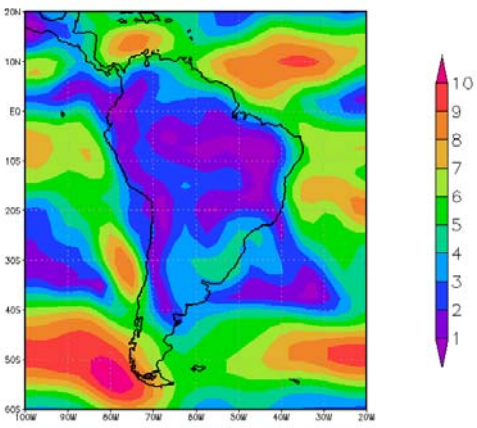
Isto pode ser feito tanto para gráficos com uma ou duas dimensões.



40. Barra de cores no GrADS

Você precisa ter alguns arquivos para que a barra colorida possa aparecer. São eles: cbar.gs (barra quadrada), cbarn.gs (barra com ponta nas extremidades) e cbarc.gs (barra no canto direito superior).

Para que a barra apareça é necessário que seu gráfico seja do tipo preenchido (shaded) ou quando plotamos a magnitude do vento através da opção vector. O comando que mostra a barra deve ser executado após a visualização da variável. Veja os exemplos abaixo.

run cbar.gs	run cbarc.gs
	
<pre>'reinit' 'sdfopen uwnd.mon.mean.nc' 'sdfopen vwnd.mon.mean.nc' 'set lat -60 20' 'set lon -100 -20' 'set gxout shaded' 'd mag(uwnd,vwnd.2)' 'cbar' ;*desenha a escala de cores</pre>	<pre>'reinit' 'sdfopen uwnd.mon.mean.nc' 'sdfopen vwnd.mon.mean.nc' 'set lat -60 20' 'set lon -100 -20' 'set gxout shaded' 'd mag(uwnd,vwnd.2)' 'cbarc' ;*desenha a escala de cores</pre>
run cbarn.gs	
	
<pre>'reinit' 'sdfopen uwnd.mon.mean.nc'</pre>	


```
'sdfopen vwnd.mon.mean.nc'
'set lat -60 20'
'set lon -100 -20'
'set gxout shaded'
'd mag(uwnd,vwnd.2)'
'cbarn' ;*desenha a escala de cores
```

41. Salvando figuras no GrADS

Após ter atribuído todas as características a sua figura, digite o comando “printim” para salvar a(s) figura(s). Este comando deve ser a última linha do seu “.gs”. Certifique-se que você tem o arquivo “gxgif” no diretório onde estão as bibliotecas do GrADS para salvar no formato “.gif”.

Exemplo:

```
'reinit'
'sdfopen uwnd.mon.mean.nc'
'set lat -60 20'
'set lon -100 -20'
'set gxout shaded'
'd uwnd'
'cbarn'
'printim vento.gif' ;*salva a figura com o nome “vento” no formato “.gif”
```

42. Criando um arquivo “.gs” no GrADS

Para gerar um arquivo “.gs” abra seu editor de texto preferido e salve o arquivo com um nome qualquer com a extensão “.gs”.

Importante: Não se esqueça de colocar os comandos do GrADS entre aspas no seu script, caso contrário seu “.gs” terá problemas e não será executado.

Exemplo de um arquivo .gs:

'reinit'	;*reinicia o grads toda vez que executa o arquivo “.gs”
'open 250304.ctl'	;*abre o arquivo 250304.ctl
'set map 1 1 6'	;*fixa cor (1), estilo (1) e espessura (6) ao mapa
'set lat -45 -10'	;*fixa a latitude
'set lon -100 -10'	;*fixa a longitude
'set lev 925'	;*fixa o nível desejado
'set grads off'	;*desabilita a propaganda do GrADS
'set grid off'	;*desabilita a grade do GrADS
'set gxout contour'	;*seleciona o gráfico do tipo contorno ou isolinhas
'set cint 2'	;*especifica o intervalo entre as isolinhas
'set cthick 6'	;*especifica a espessura da isolinha
'd temp'	;*visualiza a variável desejada (Temperatura)
'draw title Temperatura em 925 hPa'	;*escreve o título no gráfico
'printim temperatura.gif'	;*salva o arquivo com o formato “.gif”

43. Executando arquivo “.gs”

O GrADS deve estar aberto para executar o arquivo “.gs”, para isso basta digitar:

ga-> run nomedoarquivo.gs ou simplesmente **ga-> nomedoarquivo.gs**

O arquivo pode ser executado sem colocar o “run” na frente do “.gs”

44. Programas Adicionais

44.1. Usando o script grib2ctl.pl

Caso seu arquivo “.grb” não possua seu respectivo “.ctl” o script “grib2ctl.pl” facilita sua criação. Abra um terminal no Linux onde estão seus dados “.grb” e use o script para criar o “.ctl”. Veja o exemplo abaixo para melhor entendimento.

```
gui@pos-guilherme:~/grib$ ls script grib2ctl.pl
05021600.GrbF000
                                arquivo "grib"
gui@pos-guilherme:~/grib$ grib2ctl.pl 05021600.GrbF000 > 05021600.GrbF000.ctl
gui@pos-guilherme:~/grib$ ls
05021600.GrbF000 05021600.GrbF000.ctl
                                arquivo ".ctl" a ser gerado
gui@pos-guilherme:~/grib$ gribmap -i 05021600.GrbF000.ctl -0
                                gera o arquivo ".idx" com base no ".ctl"
gribmap: Scanning binary GRIB file(s):
gribmap: Opening GRIB file 05021600.GrbF000
gribmap: Reached EOF
gribmap: Writing the map...

gui@pos-guilherme:~/grib$ ls
05021600.GrbF000 05021600.GrbF000.ctl 05021600.GrbF000.idx
```

44.2. Usando o script lats4d.gs

O script “lats4d.gs” é muito útil quando se deseja “cortar” os dados para uma determinada data, área, nível, latitude, longitude etc. Isso ficará a critério do usuário fazer as devidas modificações.

Exemplo 1: Neste caso será gerado um arquivo “.grb” (-format grads_grib) com seu respectivo “.ctl”

```
Grid Analysis and Display System (GrADS) Version 1.9b4
Copyright (c) 1988-2005 by Brian Doty and IGES
Center for Ocean-Land-Atmosphere Studies (COLA)
Institute for Global Environment and Society (IGES)
GrADS comes with ABSOLUTELY NO WARRANTY
See file COPYRIGHT for more information

Config: v1.9b4 32-bit little-endian readline sdf/xdx netcdf lats printim

Issue 'q config' command for more information.

Landscape mode? (no for portrait):
GX Package Initialization: Size = 11 8.5
ga->
ga->
ga-> lats4d -i air.1980.nc -o temperatura -format grads_grib -time 01jan1980 31jan1980 -levs 1000 925 850 -lat -60 20 -lon -100 -20 -q
No hardcopy metafile open
All files closed; all defined objects released
All GrADS attributes have been reinitialized

gribmap: Scanning binary GRIB file(s):
gribmap: Opening GRIB file temperatura.grb
gribmap: Reached EOF
gribmap: Writing the map...

LATS_GRIB: SUCCESS -- gribmap for GrADS/VCS seems to have worked...

lats4d: created grads_grib file temperatura.grb
lats4d: exiting from GrADS...
No hardcopy metafile open
GX package terminated
gui@zeus:/media/GUI_500GB/dados/R2/NCEP_DIARIOS/air$
```

Diagrama de anotações para o Exemplo 1:

- arquivo de entrada: `air.1980.nc`
- arquivo de saída: `temperatura`
- formato do arquivo de saída: `grads_grib`
- tempo fixado: `01jan1980 31jan1980`
- níveis desejados: `1000 925 850`
- latitude desejada: `-60 20`
- longitude desejada: `-100 -20`
- sai do GrADS: `-q`

Exemplo 2: Neste caso será gerado apenas um arquivo “.bin” (-format stream). Este formato é muito utilizado quando se deseja usar com Fortran.

```
Grid Analysis and Display System (GrADS) Version 1.9b4
Copyright (c) 1988-2005 by Brian Doty and IGES
Center for Ocean-Land-Atmosphere Studies (COLA)
Institute for Global Environment and Society (IGES)
GrADS comes with ABSOLUTELY NO WARRANTY
See file COPYRIGHT for more information

Config: v1.9b4 32-bit little-endian readline sdf/xdx netcdf lats printim

Issue 'q config' command for more information.

Landscape mode? (no for portrait):
GX Package Initialization: Size = 11 8.5
ga->
ga->
ga-> lats4d -i air.1980.nc -o temperatura -format stream -time 01jan1980 31jan1980 -levs 925 -q
No hardcopy metafile open
All files closed; all defined objects released
All GrADS attributes have been reinitialized
lats4d: created stream file temperatura.bin
lats4d: exiting from GrADS...
No hardcopy metafile open
GX package terminated
```

Diagrama de anotações para o Exemplo 2:

- arquivo de entrada: `air.1980.nc`
- arquivo de saída: `temperatura`
- formato de saída do arquivo: `stream`
- tempo fixado: `01jan1980 31jan1980`
- nível desejado: `925`
- sai do GrADS: `-q`

44.3. Procedimentos para instalar EOF no GrADS

REQUERIMENTO: FORTRAN90 e ser root na sua máquina

1 - Neste endereço <http://www.atmos.ucla.edu/~munnich/Grads/EOF/> faça o download do arquivo para sua plataforma Linux.

2 - Crie o diretório "eof_grads" em "/usr/local/lib/grads/" e copie o arquivo "eofudf_linux.tar.gz" para o diretório criado.

3 - Vá para o diretório "eof_grads" e descompacte o arquivo fazendo "tar -xvf eofudf_linux.tar.gz". Com isso serão criados os arquivos: "eof.gs", "eofudf", "libcxa.so.3" e "readme.html". Copie o arquivo "eof.gs" para "/usr/local/lib/grads".

4 - Faça o download do código fonte (eofgrads_src.tar.gz) que se encontra no endereço do item 1 deste tutorial para "/usr/local/lib/grads/eof_grads". Descompacte este arquivo fazendo "tar -xvf eofgrads_src.tar.gz". Com isso, surge o diretório "eofgrads_v0155". Dentro dele estão os arquivos que serão compilados.

4.1 - Entre no diretório "eofgrads_v0155/src" e edite o Makefile de acordo com o compilador Fortran 90 que você usa na sua máquina. As alinhas abaixo são do arquivo "Makefile".

defaults

FC=gfortran --> mude aqui apenas o nome do compilador Fortran 90

FFLAGS=-O Ma

LDFLAGS=-O

double precision

#FFLAGS=-O -N113

#LDFLAGS=-O -N113

#Intel f90

#FC=ifc

#AIX

#FC=xlf

#FFLAGS=-O -w -axW # Intel ifc + P4 processor --> comente aqui colocadando #

#LDFLAGS=-O -w -axW # Intel ifc + P4 processor --> comente aqui colocadando #

eofudf: eofudf.o eispacks.o

\$(FC) \$(LDFLAGS) -o eofudf eofudf.o eispacks.o

jeofudf: jeofudf.o eispacks.o

\$(FC) \$(LDFLAGS) -o jeofudf jeofudf.o eispacks.o

clean:

rm -f eofudf eofudf.o *.mod eispacks.o *.d

.f.o:

`$(FC) $(FFLAGS) -c $<`

- Depois faça "make clean" e em seguida "make". Tudo isso é feito dentro de "src". Com estes passos, serão gerados os arquivos: "eispacks.o", "eofudf", "eofudf.o", "sizes.mod" e "udf.mod".

- Dentro de "/usr/local/lib/grads/" deve ter um diretório chamado "udf". Se ele não existir, faça "mkdir udf". Coloque o arquivo "eofudf" que esta em "/usr/local/lib/grads/eof_gs/eofgrads_v0155/src" dentro de "/usr/local/lib/grads/udf".

6 - Dentro de "/usr/local/lib/grads" crie o arquivo "udft.txt" e edite-o colando as seguintes linhas:

```
eofudf 2 6 expr value value value value char
sequential
/usr/local/lib/grads/udf/eofudf
eofudf.in
eofudf.out
```

7 - Vá para o seu home e faça "ls -a" e edite seu arquivo ".bashrc" e cole nele as seguintes linhas.

```
export GAUDFT=/usr/local/lib/grads/
export GAUDFT=/usr/local/lib/grads/udft.txt
```

8 - Vá para "/usr/local/lib/grads/eof_grads" e copie o arquivo "libcx.so.3" para "/usr/lib/".

9 - Com estes passos você já pode gerar seus gráficos de EOF usando o GrADS.

10 - Para fazer os gráficos execute o GrADS e abra o arquivo que desejar para fazer a EOF

```
'gradsnc'                                ;*abre o grads
'sdfopen uwnd.nc'                        ;*abre o arquivo ".nc"
'set t 1 100'                            ;*fixa 100 tempos (fica ao critério do usuário)
'set lat -40 10'                         ;*fixa latitude
'set lon 270 330'                        ;*fixa longitude
'eof -neof 12 -npc 100 -tinc 1 -nper 70 -o saida uwnd'
```

Onde:

-neof = número de EOFs (default = 12)
 -npc = número PCs (default = -neof)
 -tinc = incremento para fazer o cálculo (default = 1)
 -nper = 70 -> porcentagem de dados em cada ponto de grade para fazer a EOF
 -o = nome do arquivo de saída (fica ao critério do usuário)
 uwnd = nome da variável do arquivo (.ctl ou .nc)

11 - Se tudo ocorrer bem as linhas abaixo serão vistas.

```
ga-> eof -neof 12 -npc 100 -tinc 1 -nper 70 -o saida uwnd
Writing data to transfer file...
Executing eofudf binary ...
```

```
Number of EOFs: 12
Number of PCs: 12
Required % of defined data: 70
Output files prefix: saida_
```

```

Space dim: 1089
Time dim: 100
Max: 24.56000
Min: -18.69000
Usable space points: 1089 of 1089
Usable time slices: 100 of 100
No undefs in packed data.
Computing covariance matrix...
Total variance: 13673.64
Computing eigenvectors...
Total variance: 13673.64

```

No.	E.value	% Var
1	2270.	16.602
2	1419.	10.376
3	1165.	8.521
4	944.1	6.905
5	798.9	5.843
6	703.5	5.145
7	529.9	3.875
8	506.5	3.705
9	437.5	3.200
10	387.5	2.834
11	357.3	2.613
12	349.9	2.559

```

Writing EOFs...
Writing PCs...
Writing time info to CTLs...

```

12 - O arquivo "saida_eof.ctl" contém as PC's no espaço e o arquivo "saida_eval.txt" estão as porcentagens das PC's que foram selecionadas.

13 - A série temporal de cada PC pode ser vista no arquivo "saida_pc.ctl". Observe que cada nível "z" corresponde a cada PC. Para visualizar a primeira PC basta fazer no grads.

```

'set z 1' ;* se fosse a segunda, basta fazer 'set z 2' ... 'set z 12'
'set t 1 100'
'd pc'

```

44.4. Instalando o regrid2 no GrADS

O regrid2 serve para interpolar campos. Abaixo são mostrados os procedimentos para sua instalação.

1 – Copiar o arquivo “regrid2.tar.gz” disponível em <http://ftp.kumamoto-u.ac.jp/pub/.2/graphic/grads/sprite/udf/> para “/usr/local/lib/grads”.

2 – Dentro de “/usr/local/lib/grads” fazer “tar -zxvf regrid2.tar.gz”. Este procedimento cria o diretório “udf” em “/usr/local/lib/grads”. Dentro deste diretório encontra-se o manual do regrid2.

3 – Entrar em “udf” e ir para “regrid2”.

4 – Edite o arquivo “Makefile”. As linhas abaixo são do arquivo Makefile. Modifique as linhas em vermelho colocando o caminho onde esta o diretório grads e o tipo de compilador usado na máquina (para este exemplo esta sendo usado o g77).

```

SHELL=/bin/sh
#
#
#
REGRID_DIR = /usr/local/lib/grads

SRCS=      regrid2.f

OBJS=regrid2.o
#
CFT=      g77
LDR=      g77
CFT=      g77
LDR=      g77

CFT=      g77
LDR=      g77

LDRFLAGS= -o
#LDRFLAGS=-32 -o

EXE=      $(REGRID_DIR)/udf/regrid2.exe
EXE=      ../regrid2.exe

#FFLAGS= -c -O -32

FFLAGS= -c -O

$(EXE):    $(OBJS)
           $(LDR) $(LDRFLAGS) $(EXE) $(OBJS)
.f.o:
           $(CFT) $(FFLAGS) *.f

del_lists:
           -rm -f $(LISTS)

clean:
           -rm -f $(OBJS)

```

6 – Digitar no terminal do Linux “make clean” e depois “make -f Makefile”.

7 – Vá para seu home e edite seu arquivo “.bashrc”. Para ver este arquivo basta digitar no Linux “ls -a”. Com esta opção os arquivos ocultos (arquivos que possuem um ponto na frente do nome do arquivo) serão mostrados. Adicione a linha abaixo no seu “.bashrc”.

export GAUDFT=/usr/local/lib/grads/udf/regrid2/udft.regrid2

8 – Fazer source “.bashrc” para atualizar seu “.bashrc”.

9 – Abra o GrADS e digite “q udf”. Se tudo ocorrer bem, será mostrada uma linha dizendo onde esta instalado o regrid.

10 – Com isso, você será capaz de interpolar seus campos.

Exemplo:

```
'reinit'
'sdfopen air.nc'
'd regrid2(air,1.5,1.5,bl)      ;*"air" é a variável a ser interpolada, "1.5" corresponde a resolução
especial (em longitude e latitude) que será interpolada e "bl" é o tipo de interpolação.
```

45. Linguagem de Programação no GrADS

45.1. Operadores

	Operador lógico OU	&	Operador lógico E
!	Unary not	-	Unary minus
%	Concatenação	=	Igual
!=	Diferente	>	Maior que
>=	Maior ou igual a	<	Menor que
<=	Menor ou igual	+	Soma
-	Subtração	*	Multiplicação
/	Divisão		

45.2. Fluxo de Controle

45.2.1. Comando WHILE

O controle de fluxo “WHILE” tem diversas aplicações. Dependerá das necessidades do usuário empregá-lo corretamente para suas necessidades. Abaixo, são mostrados alguns exemplos da sua utilização.

Exemplo: Usando o "while" para varrer 10 tempos do arquivo.

```
'reinit'
'sdfopen ATSM.nc'
t1 = 1      ;* Tempo inicial que será definido pelo usuário
t2 = 10     ;* Tempo final que será definido pelo usuário
*Enquanto (while) t1 (1) for menor ou igual a t2 (10)
while(t1<=t2)
'set t 't1   ;*varre os tempos de t1 (1) até t2 (10)
'd anom'    ;*visualiza a variável (anom)
pull enter  ;*espera por um comando do usuário (PRESSIONE ENTER)
'c'         ;*limpa a tela para mostrar o próximo tempo
t1 = t1 + 1
endwhile
```


45.2.2. Comando IF

O controle de fluxo “IF” tem por finalidade substituir variáveis como também servir de condição para determinadas tarefas, dentre outras possibilidades que dependerá da imaginação do usuário. Abaixo, são mostrados alguns exemplos da sua utilização.

Exemplo1: O comando IF usado aqui, substitui a string mes (JAN, FEB, ... , DEC) do GrADS (lembrando que os meses no GrADS estão em inglês) para valor numérico (JAN = 01, FEB = 02 , ... , DEC = 12

```
'reinit'
'sdfopen ATSM.nc'
t1 = 1          ;*Tempo inicial
t2 = 50         ;*Tempo final
*Enquanto (while) t1 (1) for menor ou igual a t2 (10)
while(t1<=t2)

'set t t1        ;*varre os tempos de t1 (1) até t2 (10)

'q dims'         ;*fornece detalhes sobre o arquivo aberto

linha5 = sublin(result,5)    ;*captura a linha 5 do comando "q dims"
data  = subwrdr(linha5,6)    ;*captura a string "00Z22FEB1985" do comando "q dims"
dia   = substr(data,4,2)     ;*captura o dia da string "00Z22FEB1985"
mes   = substr(data,6,3)     ;*captura o mes da string "00Z22FEB1985"
ano   = substr(data,9,4)     ;*captura o ano da string "00Z22FEB1985"
hora  = substr(data,1,2)     ;*captura a hora da string "00Z22FEB1985"

*troca os meses JAN, FEB, ... , DEC para valores numéricos, 01, 02, ... , 12

if(mes=JAN);mesc=01;endif
if(mes=FEB);mesc=02;endif
if(mes=MAR);mesc=03;endif
if(mes=APR);mesc=04;endif
if(mes=MAY);mesc=05;endif
if(mes=JUN);mesc=06;endif
if(mes=JUL);mesc=07;endif
if(mes=AUG);mesc=08;endif
if(mes=SEP);mesc=09;endif
if(mes=OCT);mesc=10;endif
if(mes=NOV);mesc=11;endif
if(mes=DEC);mesc=12;endif

'd anom'         ;*visualiza a variável (anom)
'draw title Anomalia da TSM no dia 'dia'/'mesc'/'ano
pull enter       ;*espera por um comando do usuário (PRESSIONE ENTER)
'c'              ;*limpa a tela
t1 = t1 + 1
endwhile
```

Exemplo2: Neste exemplo são mostradas as aplicações dos comandos WHILE e IF. Serão abertos vários arquivos (1985 a 1990) e em seguida 5 tempos de cada arquivo aberto são visualizados e após a visualização dos tempos o mesmo é fechado para a abertura do próximo.

```

'reinit'
'set display color white'
'c'

f1 = 1985
f2 = 1990

while(f1<=f2)
'sdfopen ATSM.'f1'.nc'      ;*abre a sequência de arquivos

t1 = 1      ;* Tempo inicial
t2 = 5      ;* Tempo final

*varre os tempos de t1 (1) até t2 (5)
while(t1<=t2)

'set t 't1
'set lat -60 10'
'set lon -90 -20'

'q dims'      ;*fornece detalhes do arquivo aberto

linha5 = sublin(result,5)      ;*captura a linha 5 do comando "q dims"
data = subwrdr(linha5,6)      ;*captura a string "00Z22FEB1985" do comando "q dims"
dia = substr(data,4,2)      ;*captura o dia da string "00Z22FEB1985"
mes = substr(data,6,3)      ;*captura o mês da string "00Z22FEB1985"
ano = substr(data,9,4)      ;*captura o ano da string "00Z22FEB1985"
hora = substr(data,1,2)      ;*captura a hora da string "00Z22FEB1985"

if(mes=JAN);mesc=01;endif
if(mes=FEB);mesc=02;endif
if(mes=MAR);mesc=03;endif
if(mes=APR);mesc=04;endif
if(mes=MAY);mesc=05;endif
if(mes=JUN);mesc=06;endif
if(mes=JUL);mesc=07;endif
if(mes=AUG);mesc=08;endif
if(mes=SEP);mesc=09;endif
if(mes=OCT);mesc=10;endif
if(mes=NOV);mesc=11;endif
if(mes=DEC);mesc=12;endif

'd anom'      ;*visualiza a variável (anom)
'draw title Anomalia da TSM no dia 'dia'/'mesc'/'ano'      ;*escreve o título no gráfico

pull enter      ;*espera por um comando do usuário (PRESSIONE ENTER)
'c'
t1 = t1 + 1
endwhile

'reinit'      ;*fecha o arquivo
'set display color white'
'c'
f1 = f1 + 1
endwhile

```

46. Salvando arquivo texto no GrADS

Esta opção é útil quando se deseja salvar uma determinada série de dados para ser usada em outro pacote gráfico para visualização (SURFER, MATLAB, STATISTICA etc.) ou até mesmo para manipulação usando a Linguagem FORTRAN. Veja o exemplo abaixo.

```
'reinit'
'sdfopen ATSM.nc'
t1 = 1          ;* Tempo inicial
t2 = 10         ;* Tempo final
*Enquanto (while) t1 (1) for menor ou igual a t2 (10)
while(t1<=t2)
'set t 't1      ;*varre os tempos de t1 (1) até t2 (10)
'set lat 0'     ;* fixa a latitude no Equador
'set lon -20'   ;* fixa a longitude em 120W (longitudes oeste = negativas)

'q dims'        ;*fornece detalhes sobre o arquivo aberto

linha5 = sublin(result,5) ;*captura a linha 5 do comando "q dims"
data = subwrđ(linha5,6) ;*captura a string "00Z22FEB1985" do comando "q dims"
dia = substr(data,4,2) ;*captura o dia da string "00Z22FEB1985"
mes = substr(data,6,3) ;*captura o mes da string "00Z22FEB1985"
ano = substr(data,9,4) ;*captura o ano da string "00Z22FEB1985"
hora = substr(data,1,2) ;*captura a hora da string "00Z22FEB1985"

if(mes=JAN);mesc=01;endif
if(mes=FEB);mesc=02;endif
if(mes=MAR);mesc=03;endif
if(mes=APR);mesc=04;endif
if(mes=MAY);mesc=05;endif
if(mes=JUN);mesc=06;endif
if(mes=JUL);mesc=07;endif
if(mes=AUG);mesc=08;endif
if(mes=SEP);mesc=09;endif
if(mes=OCT);mesc=10;endif
if(mes=NOV);mesc=11;endif
if(mes=DEC);mesc=12;endif

'd anom'       ;*visualiza a variável "anom"
```

*Observe que foi selecionado apenas um ponto (set lat 0 e set lon -20). E o tempo esta variando. Logo, o resultado será uma série temporal.

```
                                posições      1   2   3   4
atasm = subwrđ(result,4)      ;*captura a posição 4 de "Result value = 0.4"

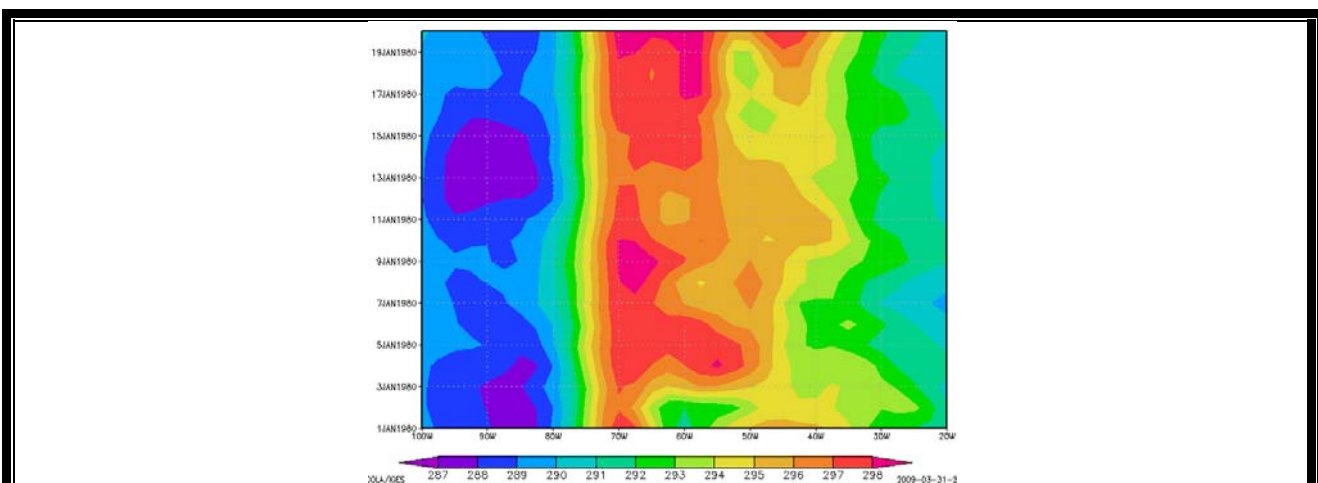
*será escrito no arquivo de saída "val_tsm.dat":
*dia = coluna 1, mês = coluna 2, ano = coluna 3 e Anomalia da TSM = coluna 4
saida = write(val_tsm.dat,dia' 'mesc' 'ano' 'atasm)
say dia' 'mesc' 'ano' 'atasm ;*mostra na tela o que será guardado no arquivo ".dat"
t1 = t1 + 1
endwhile
```

47. Aplicações diversas

47.1. Gráfico de linha de uma área

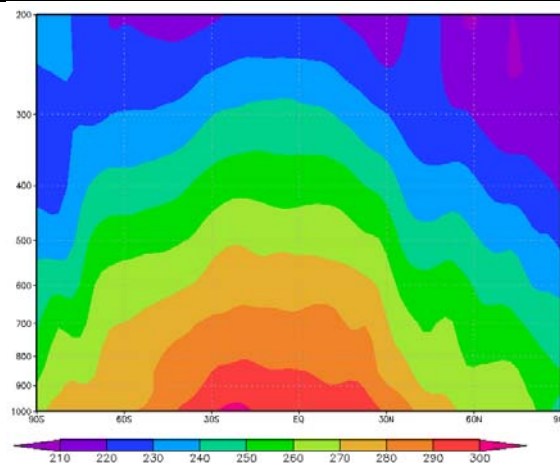
```
'reinit'
'sdfopen air.1980.nc'
'set lat 0'      ;*tem que fixar uma latitude qualquer. Ela não terá nenhuma influência na figura
'set lon -60'    ;*tem que fixar uma longitude qualquer. Ela não terá nenhuma influência na figura
'set t 1 20'     ;*tempo que será visualizado
*é necessário fixar um ponto (lat,lon) para a função "tloop" funcionar. A função "aave" realiza o
cálculo de média (tempo, latitude, longitude etc.) e retorna apenas um valor
'd tloop(aave(air,lon=-60,lon=-55,lat=-5,lat=-2.5))'
'printim graf_linha.gif'      ;*salva a figura no formato ".gif"
```

47.2. Gráfico Hovmoller (x = longitude e y = tempo)



```
'reinit'
'sdfopen air.1980.nc'
'set lat 0'      ;*tem que fixar uma latitude qualquer. Ela não terá nenhuma influência na figura
'set lon -100 -20' ;*faixa de longitude desejada
'set lev 925'     ;*fixa o nível desejado
'set t 1 20'      ;*tempo que será visualizado
'set gxout shaded' ;*gráfico do tipo preenchido
'd ave(air,lat=-20,lat=-10)' ;*faz média (ave) na latitude de -20 a -10
'cbarn'           ;*mostra a escala de cores
'printim hov_xlon_ytempo.gif' ;*salva a figura no formato ".gif"
```

47.3. Gráfico Hovmoller (x = latitude e y = altura)



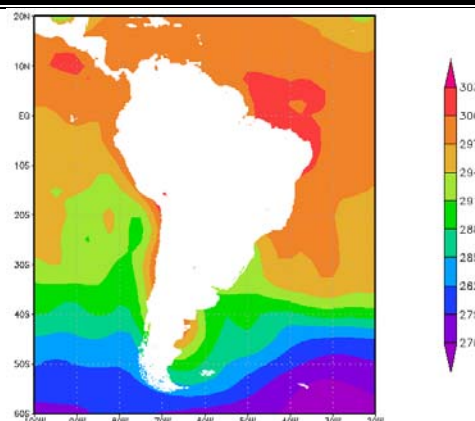
```
'reinit'
'set display color white'
'c'
'sdfopen air.nc'
'set t 20' ;*tempo que será feito o gráfico
'set lon -60' ;*tem que fixar uma latitude qualquer. Ela não terá nenhuma influência na figura
'set lev 1000 200' ;*nível desejado
'set grads off'
'set gxout shaded'
'set zlog on' ;*habilita escala logarítmica
'd ave(air,lon=-60,lon=-50)' ;*faz média na longitude
'cbarn'
'printim hov_xlat.yaltura.gif'
```

47.4. Mascarando continente ou oceano

Para plotar apenas os valores sobre o continente ou oceano, será usado o basemap.gs.

IMPORTANTE: Os arquivos: lpoly_mres.asc, lpoly_US.asc, lpoly_US.asc.part, lpoly_hires.asc, lpoly_lowres.asc e basemap.gs devem estar no mesmo diretório onde se encontra o seu “.gs” do contrário, ele não funcionará.

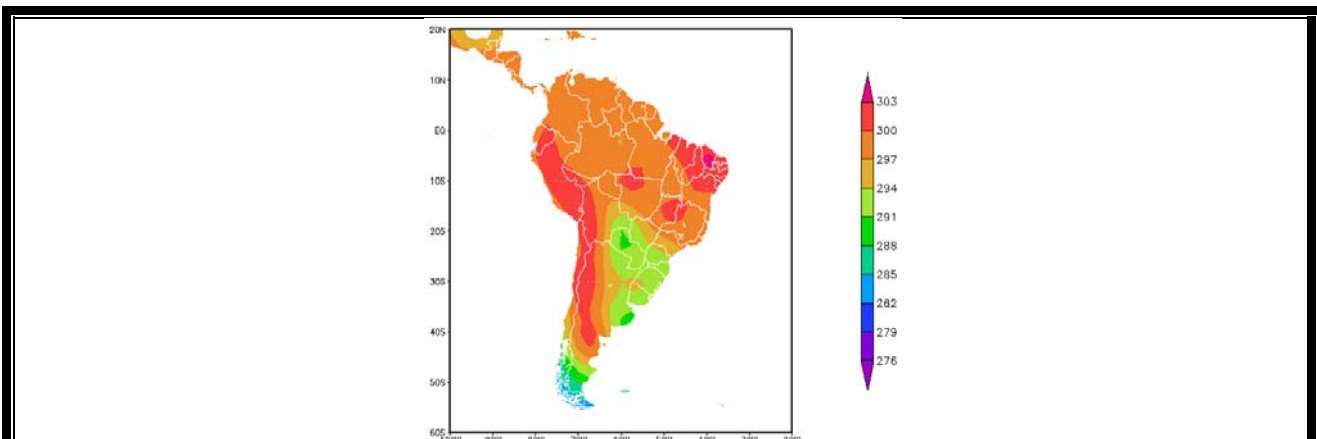
Exemplo1: Mascarando apenas o continente



```
'reinit'
'set display color white'
'c'
'sdfopen air.1980.nc'
'set lat -60 20'
'set lon -100 -20'
'set t 1'
'set grads off'
'set gxout shaded'
'd air'
'basemap L 0 0 M'
'cbarn'
'printim basemap_mask_conti.gif'
```

*O argumento L = diz para mascarar o continente. Se fosse para mascarar o oceano, mudar de "L" para "O". O segundo argumento refere-se à cor do preenchimento (ver tabela de cores). O terceiro argumento define uma cor para o contorno do preenchimento (tabela de cores) e o quarto argumento, desenha ou não desenha os Estados e os países e pode assumir os valores: M, L ou H. Neste caso, mascaramos o continente (L). Definimos a cor branca tanto para o preenchimento (0) como para o contorno (0) e o último argumento (M) não interfere no gráfico.

Exemplo2: Mascarando apenas o oceano



```
'reinit'
'set display color white'
'c'
'sdfopen air.1980.nc'
'set lat -60 20'
'set lon -100 -20'
'set grads off'
'set gxout shaded'
'set mpdset brmap' ;*divide os Estados do Brasil e o países
'd air'
'basemap O 0 0 L'
'cbarn'
'printim basemap_mask_oceano.gif'
```

*mascarando apenas o oceano (O). Selecionamos a cor branca para representar o oceano (0) e o contorno do continente (0).

47.5. Usando o script “color.gs” para criar escala de cores personalizada

Este “.gs” pode ser encontrado no link número 7 que esta no final da apostila.

```
'reinit'
'set display color white'
'c'
'sdfopen air.1980.nc'
'set lat -60 20'
'set lon -100 -20'
'set grads off'
'color -gxout shaded -kind white->gray->black -var air-273.15 5 30 5'
'd air-273.15'
'cbarn'
```

O script color.gs cria uma escala de cores definida pelo usuário. Dentro deste script estão as cores disponíveis. Fica ao critério do usuário escolher a melhor cor para o seu gráfico. Veja que white, gray e Black são cores que estão disponíveis dentro do script. A variável usada foi a temperatura do ar (air), sendo que ela foi transformada de Kelvin para Celsius. Os demais valores, 5 30 5 representam o menor valor, maior valor e o intervalo de variação, respectivamente.

47.6. Usando o script “clave.gs” para gerar climatologia

Este “.gs” pode ser encontrado no link de número 5 que esta no final da apostila.

Exemplo1: Climatologia para DEZ-JAN-FEV

```
'reinit'
'set display color white'
'c'
'sdfopen air.nc'
'set lat -60 20'
'set lon -100 -20'
'set grads off'
'set gxout shaded'
'set lev 1000'
'clave air 01dec%y 28feb%ypp 1979 1986'
'cbarn'
```

A climatologia da Temperatura do Ar foi feita para o nível de 1000 hPa apenas para o verão (DEZ-JAN-FEV). Observe que tem um “pp” após o “y” em “feb”. Isto quer dizer que, quando se faz a climatologia onde se “quebra” o mês de ano para o outro ano, deve-se colocar o “pp” após o “y”. Em outras palavras, aqui é feito de 01dez/1979 até 28fev/1980, 01dez/1980 até 28fev/1981 ... 01dez/1986 até 28fev/1987 , veja que houve uma quebra no ano, por isso, usa-se o “pp”

Exemplo2: Climatologia para JUN-JUL-AGO

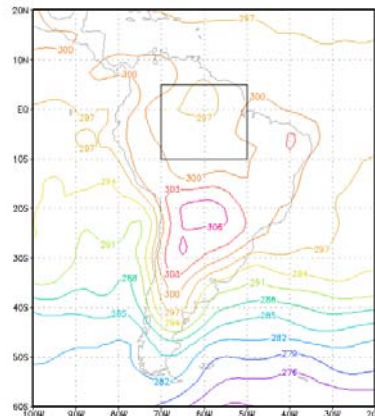
```
'reinit'
'set display color white'
'c'
'sdfopen air.nc'
'set lat -60 20'
'set lon -100 -20'
'set grads off'
```

```
'set gxout shaded'
'set lev 1000'
'clave air 01jun%y 31jul%y 1979 1986'
'cbarn'
```

Observe que não tem "pp" pois não houve quebra de ano. Observe que foi feito de 01jun/1979 até 31jul1979, 01jun/1980 até 31jul1980 ... 01jun/1986 até 31jul1986.

47.7. Usando o script “drawbox.gs” para desenhar uma caixa

Este “.gs” pode ser encontrado no link número 5 que esta no final da apostila.

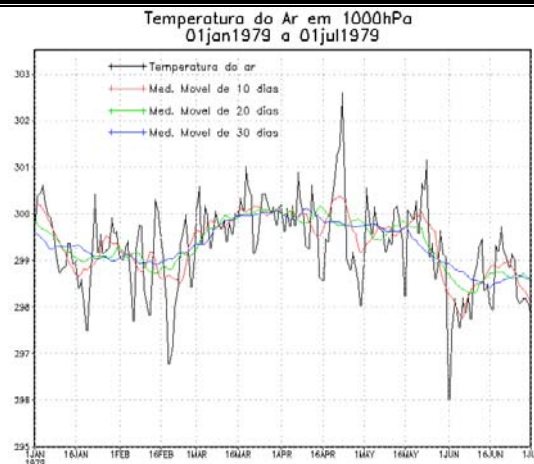


```
'reinit'
'set display color white'
'c'
'sdfopen air.nc'
'set lat -60 20'
'set lon -100 -20'
'set grads off'
'set gxout shaded'
'd air'
'drawbox -60 -50 -10 0'
'printim caixa.gif'
```

Desenha uma caixa, dado as coordenadas lon1 lon2 lat1 lat2. Onde: lon1 = longitude mais oeste lat2 = latitude mais sul. Em outras palavras, foi desenhada uma caixa que abrange as longitudes de 60W a 50W e latitudes de -10 a 0.

47.8. Usando o script “rmean.gs” para calcular média móvel

Este “.gs” pode ser encontrado no link número 3 que esta no final da apostila.



```
'reinit'
'set display color white'
'c'
'sdfopen air.nc'
'set t 1 182'
'set lon -50'
'set lat -10'
'set lev 1000'
'set grads off'
'set cmark 0' ;*define o tipo de marcador (não desenha marcador, valor 0)
'd air'
'draw title Temperatura do Ar em 1000hPa \ 01jan1979 a 01jul1979'
'set ccolor 2' ;*define cor vermelha (2)
'set cmark 0'
'rmean air 10 display' ;*realiza média móvel de 10 dias
'set ccolor 3' ;*define cor verde (3)
'set cmark 0'
'rmean air 20 display' ;*realiza média móvel de 20 dias
'set ccolor 4' ;*define cor vermelha (4)
'set cmark 0'
'rmean air 30 display' ;*realiza média móvel de 30 dias
'cbar_line -x 3.31094 -y 7.46761 -c 1 2 3 4 -m 1 1 1 1 -l 1 1 1 1 -t "Temperatura do ar" "Med.
Movel de 10 dias" "Med. Movel de 20 dias" "Med. Movel de 30 dias"
'printim rmean.gif'
```

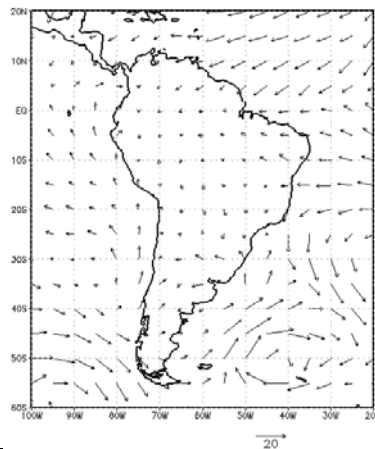
O script `cbar_line.gs` cria uma legenda de acordo com as configurações dadas pelo usuário. Os valores `-x 3.31094` e `-y 7.46761` são as posições na página (estes valores foram obtidos usando o comando `"q pos"` visto no início da apostila) onde ficarão as legendas. O `-c` representa a cor atribuída a cada legenda. O `-m` é o tipo de marcador e `-t`, os textos que serão escritos na legenda.

47.9. Convertendo a Temperatura de Kelvin para Celsius

```
'reinit'
'set display color white'
'c'
'sdfopen air.nc'
'set lon -100 -20'
'set lat -60 20'
'set grads off'
'set gxout shaded'
'd air-273.15' ;*converte de K para °C
```

47.10. Configurando os vetores no gráfico de vento

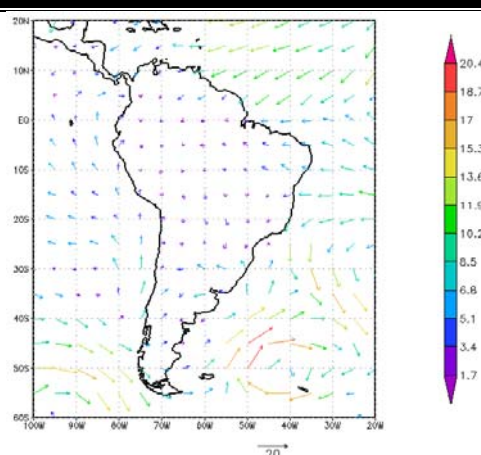
Exemplo1: Plotando somente os vetores



```
'reinit'
'set display color white'
'c'
'sdfopen uwnd.1990.nc'
'sdfopen vwnd.1990.nc'
'set lon -100 -20'
'set lat -60 20'
'set grads off'
'set map 1 1 6'
'set gxout shaded'
'd skip(uwnd,2,2);vwnd.2'
'cbarn"printim skip1.gif'
```

O argumento (2,2) após uwnd aumenta ou diminui o espaçamento entre as setas. Modifique de acordo com as suas necessidades.

Exemplo2: Plotando os vetores e suas magnitudes

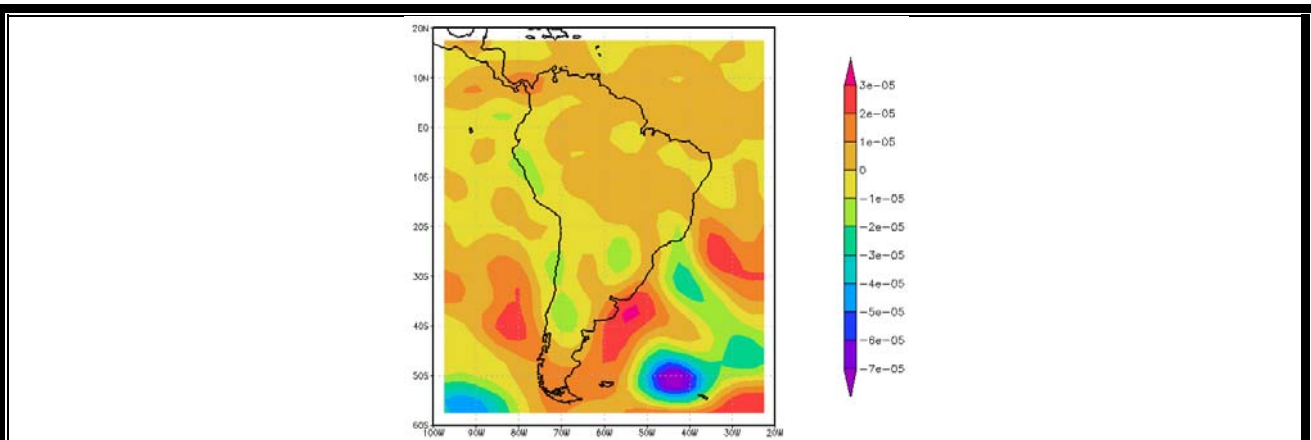


```
'reinit'
'set display color white'
'c'
'sdfopen uwnd.1990.nc'
```

```
'sdfopen vwnd.1990.nc'
'set lon -100 -20'
'set lat -60 20'
'set grads off'
'set map 1 1 6'
'set gxout shaded'
'd skip(uwnd,2,2);vwnd.2;mag(uwnd,vwnd.2)'
'cbarn'
'printim skip2.gif'
```

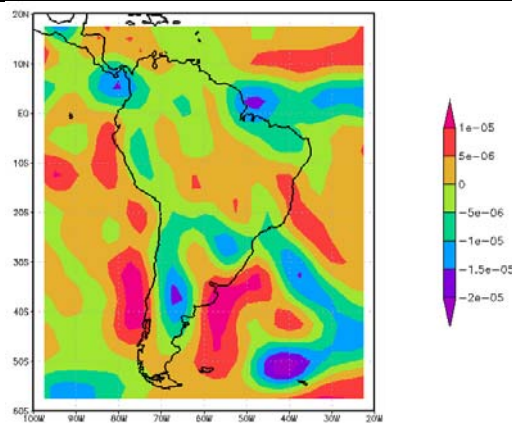
*observe que existe um novo argumento, o “mag(uwnd,vwnd.2)”. Além de plotar os vetores, nas setas serão também mostradas as magnitudes do vento que podem ser observadas na escala de cores

47.11. Calculando a vorticidade



```
'reinit'
'set display color white'
'c'
'sdfopen uwnd.1990.nc'
'sdfopen vwnd.1990.nc'
'set lon -100 -20'
'set lat -60 20'
'set grads off'
'set map 1 1 6'
'set gxout shaded'
'd hcurl(uwnd,vwnd.2)'
'cbarn'
'printim hcurl.gif'
```

47.12. Calculando a divergência



```
'reinit'
'set display color white'
'c'
'sdfopen uwnd.1990.nc'
'sdfopen vwnd.1990.nc'
'set lon -100 -20'
'set lat -60 20'
'set grads off'
'set map 1 1 6'
'set gxout shaded'
*valores positivos = divergência
*valores negativos = convergência
'd hdivg(uwnd,vwnd.2)'
'cbarn'
'printim hdivg.gif'
```

48. Links interessantes sobre GrADS

Não deixe de acessar a página oficial do GrADS. Estão disponíveis tutoriais, exemplos, scripts dentre outras ferramentas.

1. Endereço que contém esta apostila: <http://jgmsantos.googlepages.com/index.html>
2. Página oficial do GrADS: <http://www.grads.iges.org/grads>
3. Vários scripts1: http://www.atmos.umd.edu/~bguan/grads/GrADS_Scripts.html
4. Vários scripts2: <http://ccr.aos.wisc.edu/model/processing/>
5. Vários scripts3: <http://www.cptec.inpe.br/ManualGrADS/library.html>
6. Vários scripts4: http://www.ifm-geomar.de/index.php?id=grads_tools
7. Vários scripts5: <http://wind.geophys.tohoku.ac.jp/~kodamail/wiki/wiki.cgi/GrADS/script?lang=en>