

# 地球科学学院大气科学系《诊断分析与绘图实验》报告

## 实验二 数组维数、下标和变量属性

姓名	学号	成绩
马群	20201170333	100

### 一、目的：

掌握 HDF 和 GRIB 文件的读取方法；掌握维数的相关操作；学会创建 nc 格式的文件。

### 二、方法：（见实验指导书）

三、回答习题（可逐题回答，也可以把执行的命令或脚本一次写完，把要说明的内容加成注释或在最后说明）：

codes	instructions
<pre>begin f = addfile("/home/xiaoma/nc/0404/3 B43.HDF","r") pre = f-&gt;precipitation</pre>	Question1: 读取 hdf 文件并读入变量 precipitation。
<pre>printVarSummary(pre)</pre>	 <p>可以看到 hdf 中数据维数信息和 nc 文件并不相同，没有时间维且经纬度在维数上的顺序相反，同时没有单位等信息。</p>
<pre>;wks = gsn_open_wks("png","p1") ;res = True ;plot = gsn_csm_contour(wks, pre, res)</pre>	 <p>可以判断图像并不正确，由数据信息可</p>

	知经度应为横坐标，此图相反。
<pre> print(f)  pres = pre(nlat :,nlon :) lat = ispan(0,400-1,1)*0.25 - 49.875 lon = ispan(0,1440-1,1)*0.25 - 179.875 ;lat = fspan(-49.875, 49.875, 400) ;lon = fspan(- 179.875, 179.875, 1440)  lon@long_name = "longitude" lat@long_name = "latitude"  lon@units = "degrees_east" lat@units = "degrees_north" pres!0 = "lat" pres!1 = "lon" pres&amp;lat = lat pres&amp;lon = lon printVarSummary(pres)  ;wks = gsn_open_wks("png","p2") ;res = True ;plot = gsn_csm_contour(wks, pres, res) </pre>	<pre> LatitudeResolution=0.25; LongitudeResolution=0.25; NorthBoundingCoordinate=50; SouthBoundingCoordinate=-50; EastBoundingCoordinate=180; WestBoundingCoordinate=-180; Origin=SOUTHWEST; dimensions:     nlon = 1440     nlat = 400     fakeDim2 = 0     fakeDim3 = 0     fakeDim4 = 0 // unlimited </pre> <p>由数据说明信息，可知经度范围(-180°, 180°)、纬度范围(-50°, 50°)，将pre变量中的经纬度互换并根据数据信息赋予相对应数值及单位即可。</p> <p>值得一提的是 fspan() 和 ispan() 两个函数，课堂讲的是 fspan()，第三个参数为生成数据的个数；官网对于 hdf 数据的处理使用的是 ispan()，第三个参数为相邻数据的间隔，本文中两种函数均有测试，最后结果都是相同的。</p> <pre> Variable: pre Type: float Total Size: 2304000 bytes           576000 values Number of Dimensions: 2 Dimensions and sizes:  [nlon   1440] x [nlat   400] Coordinates: Number Of Attributes: 2   units :      mm/hr   hdf_name :    precipitation  Variable: pres Type: float Total Size: 2304000 bytes           576000 values Number of Dimensions: 2 Dimensions and sizes:  [lat   400] x [lon   1440] Coordinates:           lat: [-49.875..49.875]           lon: [-179.875..179.875] Number Of Attributes: 2   hdf_name :    precipitation   units :      mm/hr </pre> <p>(处理前后数据信息)</p>

	<div data-bbox="898 280 1241 645" data-label="Figure"></div> <div data-bbox="844 712 1324 797" data-label="Text"><p>可以看到在处理数据后，绘制图像的经纬度回归正常。</p></div>
<pre>;;;;;;;;;;;;;; question2 f1 = addfile("/home/xiaoma/nc/0404/0 710. grib", "r")</pre>	<div data-bbox="844 819 1230 898" data-label="Text"><p>Question2: 用 addfile 函数加载 grib 文件。</p></div>
<pre>;print(f1) levels = f1-&gt;lv_ISBL1 ;print(levels) tim1 = f1-&gt;initial_time0_hours ;print(tim1) tim2 = f1-&gt;initial_time0 ;print(tim2) T = f1-&gt;T_GDS0_ISBL(200, 3, :, :)  wks = gsn_open_wks("png", "p3") res = True plot = gsn_csm_contour(wks, T, res) end</pre>	<div data-bbox="844 1008 1339 1176" data-label="Text"><pre>variables: float T_GDS0_ISBL ( initial_time0_hours, lv_ISBL1, g0_lat_2, g0_lon_3 ) center :      European Center for Medium-Range Weather Forecasts (RSMC)  long_name :    Temperature units :        K fillvalue :    1e+20 level_indicator :  100 gds_grid_type :  0 parameter_table_version :  128 parameter_number :  130 forecast_time :  0 forecast_time_units :  hours</pre></div> <div data-bbox="844 1227 1324 1357" data-label="Text"><p>阅读数据信息，找到温度变量，从题目要求可知要确定时间和压强高度，即前两个维度。</p></div> <div data-bbox="844 1368 1339 1861" data-label="Text"><pre>Variable: levels Type: integer Total Size: 36 bytes           9 values Number of Dimensions: 1 Dimensions and sizes: [lv_ISBL1   9] Coordinates:           lv_ISBL1: [500..850] Number Of Attributes: 2   long_name :    isobaric level   units :        hPa (0)      500 (1)      600 (2)      650 (3)      700 (4)      750 (5)      775 (6)      800 (7)      825 (8)      850</pre></div> <div data-bbox="844 1879 1324 2007" data-label="Text"><p>压强高度较好确定，读出数据再 print 可知 700hPa 应为 3。 对于时间，因时间并非按照日常使用格</p></div>

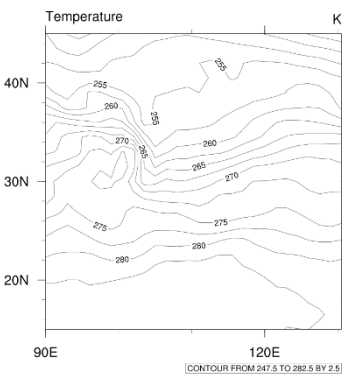
式存储，直接读取并 print 并不能直接确定题目日期对应时间，如下图。

```
Variable: tim1
Type: double
Total Size: 6760 bytes
      845 values
Number of Dimensions: 1
Dimensions and sizes:  [initial_time0_hours | 845]
Coordinates:
      initial_time0_hours: [1814520..1852440]
Number Of Attributes: 2
      long_name :   initial time
      units :      hours since 1800-01-01 00:00
(0)      1814520
(1)      1814544
(2)      1814568
(3)      1814592
(4)      1814616
(5)      1814640
(6)      1814664
(7)      1814688
(8)      1814712
(9)      1814736
(10)     1814760
(11)     1814784
(12)     1814808
```

通过阅读数据中其他信息，找到与时间相对应的字符串，读取并输出，找到对应时间，则维数信息可知。

```
(196)    01/16/2008 (00:00)
(197)    01/17/2008 (00:00)
(198)    01/18/2008 (00:00)
(199)    01/19/2008 (00:00)
(200)    01/20/2008 (00:00)
(201)    01/21/2008 (00:00)
(202)    01/22/2008 (00:00)
(203)    01/23/2008 (00:00)
(204)    01/24/2008 (00:00)
```

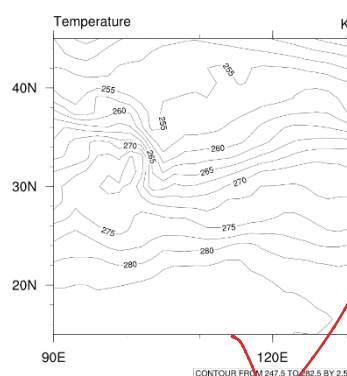
故最终绘图所需温度数据可以读出。



```
;;;;;;;;;;;;; question3
out = addfile("test.nc","c")
out->T_GDS0_ISBL = T
f2 = addfile("test.nc","r")
```

Question3:  
检验：只需读取生成的 nc 文件并绘图，可以看出可 question2 绘图是一致的，说明数据存储没有问题。

```
tes = f2->T_GDS0_ISBL
wks = gsn_open_wks("png", "p4")
res = True
res@gsnAddCyclic = False
plot = gsn_csm_contour(wks,
tes, res)
```



#### 四、实验小结（本次实验收获的经验、教训、感受等）：

本次实验报告换了一种方式呈现，相比上次还是更加美观，但空间利用率不高，下次计划改成一列表格，效果应该会好一些。

通过本次实验可以看出，不同的数据类型可以相互转化，相比 python 来说常见的气象绘图使用 ncl 还是简单些，掌握此技能后可以有更多的方式应对气象绘图（在做的大创就是因为数据是 asc 文件故选择了 python，上一个大创更是直接用了 matlab 绘图 hhh，确实是美观程度不够）。

不过 python 也有它的优势，像是本实验读数据总是会因为头文件的存在提示六行 warning，虽不影响操作，但 python 可以添加 skiprows=6 来直接跳过前六行。虽然 ncl 的 pyNgl 和 pyNio 库也不再维护，但据我了解他们是转向开发了新库 GeoCAT，不过目前国内用的好像还不多，也没有相应教程，如果可以跟进那就太好了 hhhh

你跟着看，有啥新进展跟我分享一下。