

In [6]:

```
## open file and read variables
f      = Nio.open_file(ffile, "r")          ## open data file
temp   = f.variables["tsurf"][0,:,:]        ## first time step
lat    = f.variables["lat"][:]              ## all latitudes
lon    = f.variables["lon"][:]              ## all longitudes

tempac,lon = Ngl.add_cyclic(temp,lon)

## open a workstation
wkres   = Ngl.Resources()                   ## generate an res object
wks_type = "png"                           ## output type
wks_name = "NUG_rectilinear_contour_PyNGL"
wks      = Ngl.open_wks(wks_type,wks_name,wkres)

## set resources
res      = Ngl.Resources()                   ## generate an resource object for plot

if hasattr(f.variables["tsurf"],"long_name") and hasattr(f.variables["tsurf"],"units"):
    res.tiMainString = "{} ({}).format(f.variables["tsurf"].long_name,
                                         f.variables["tsurf"].units)    ## set main title

res.cnFillOn      = True                    ## turn on contour fill.
res.cnLinesOn     = False                   ## turn off contour lines
res.cnLineLabelsOn = False                  ## turn off line labels.
res.cnInfoLabelOn = False                   ## turn off info label.
res.cnLevelSelectionMode = "ManualLevels"  ## select manual level selection mode
res.cnMinLevelValF = minval                 ## minimum contour value
res.cnMaxLevelValF = maxval                 ## maximum contour value
res.cnLevelSpacingF = inc                   ## contour increment
res.cnFillPalette  = "rainbow"              ## choose color map

res.mpGridSpacingF = 30                    ## map grid spacing

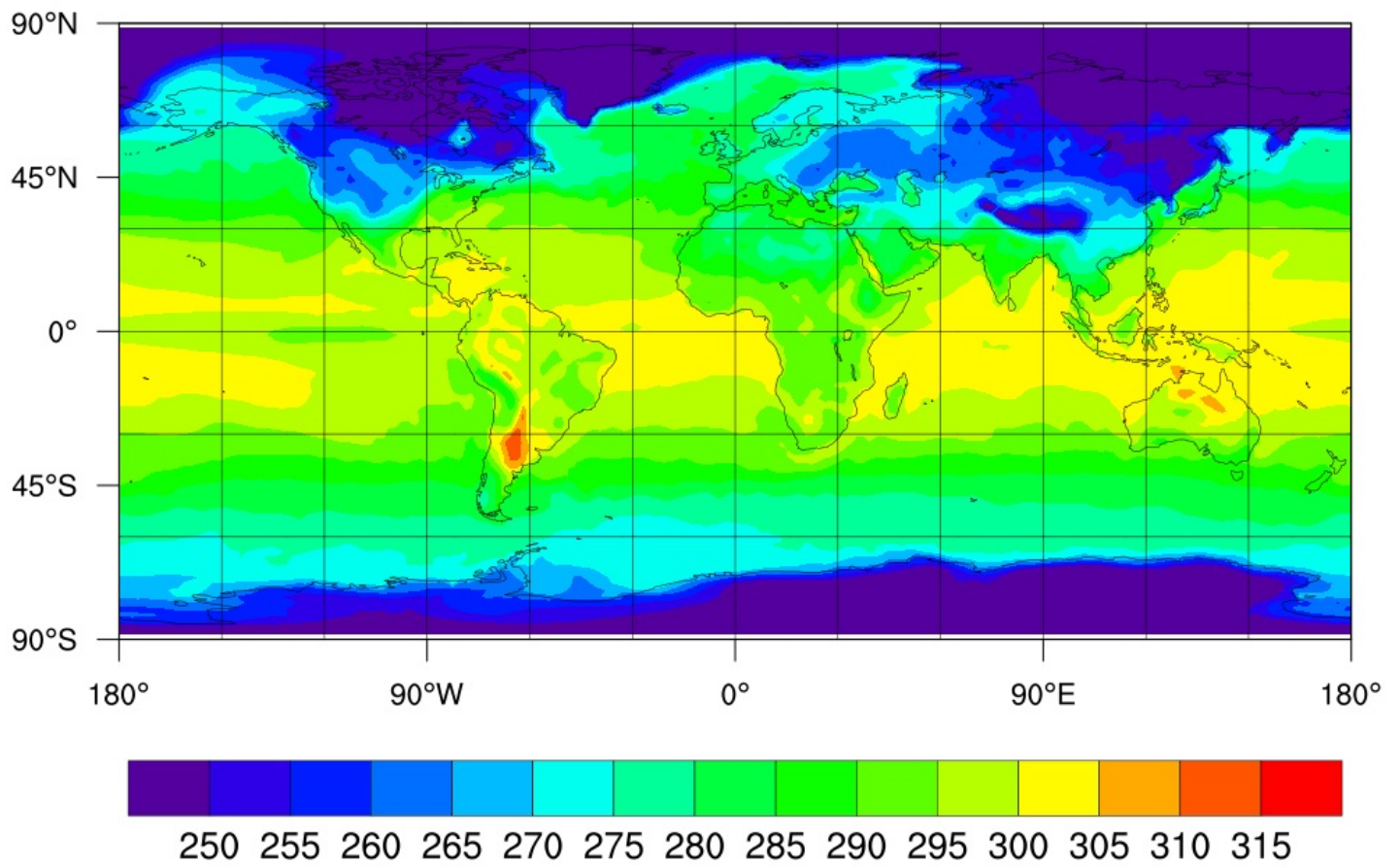
res.sfXArray       = lon                   ## longitude locations of data
res.sfYArray       = lat                   ## latitude locations of data

res.lbOrientation  = "Horizontal"          ## labelbar orientation

map = Ngl.contour_map(wks,tempac,res)      ## draw contours over a map.

## end
Ngl.end()
```

surface temperature (K)



In []: