```
;read_swath_l1_l2cc_airs.pro
;
; Reader for AIRS Radiances Swath files,
; Handles both L1B AIRS Radiance and L2 AIRS Cloud-Cleared Radiance Products
; All data fields are not read by this example.
; Add keyword=parameter to install more data fields.
;
; A keyword cannot be a substring of another keyword.
; For example, Tsurf could not be used as a keyword if
; TSurfErr is also a keyword.
;
; If reading the L2 Cloud-Cleared Radiances, the user is
; strongly advised to also read the RetQAFlag and test
; for valid data
;

pro read_swath_l1_airs, pattern, numfp, numline, $
     tai, lat, lon, rad, solzen, $
     RetQAFlag=RetQAFlag, freq=freq, topog=topog


filename = findfile ( pattern, count = cnt )
if cnt ne 1 then begin
   print, 'findfile did not return exactly one file ', cnt, pattern
   if cnt eq 0 then return
endif

print, 'read_swath_l1_airs: ', filename

FAIL = -1
SWATH_L1B_AIRS = 'L1B_AIRS_Science'
SWATH_L2CC_AIRS = 'L2_Standard_cloud-cleared_radiance_product'

fid=EOS_SW_OPEN(filename[0], /READ)
if fid eq FAIL then return

SWid=EOS_SW_ATTACH(fid, SWATH_L1B_AIRS)
if SWid eq FAIL then begin
  SWid=EOS_SW_ATTACH(fid, SWATH_L2CC_AIRS)
  If Swid eq FAIL then return
  cloud_cleared = 1
  print, "Reading "+SWATH_L2CC_AIRS
endif else begin
  cloud_cleared = 0
  print, "Reading "+SWATH_L1B_AIRS
endelse


; read required fields which are common to both
; L1B AIRS Radiance Product and L2CC AIRS Radiance Product

ret = EOS_SW_READFIELD(SWid,"Time",tai)
sz=size(tai)
numfp=sz[1]
numline=sz[2]

ret = EOS_SW_READFIELD(SWid,"Latitude",lat)
ret = EOS_SW_READFIELD(SWid,"Longitude",lon)
ret = EOS_SW_READFIELD(SWid,"radiances",rad)
ret = EOS_SW_READFIELD(SWid,"solzen",solzen)


;  read optional fields common to both L1B AIRS and L2CC AIRS products
```

```
if arg_present(topog) then ret = EOS_SW_READFIELD(SWid,"topog",topog)


; read optional fields (keyworded) which are in L1B AIRS product

if cloud_cleared eq 0 then begin   ; reading L1B AIRS Radiance Product
  if arg_present(freq) then ret = EOS_SW_READFIELD(SWid,"spectral_freq", freq )


endif else begin                     ; reading L2CC AIRS Radiance Product


; read optional fields (keyworded) which are in L2CC AIRS product

if cloud_cleared eq 1 then begin
  if arg_present(freq) then ret = EOS_SW_READFIELD(SWid,"freq", freq )
  if arg_present(RetQAFlag) then ret = EOS_SW_READFIELD(SWid,"RetQAFlag", RetQAFlag)
endif


endelse                  ; finished reading L2CC AIRS product



print, "done reading - detaching and closing the file"
ret = EOS_SW_DETACH(SWid)
ret = EOS_SW_CLOSE(fid)

end
```