
Subject: Simple Diagramming Program for IDL & PV-Wave Language Programs
Posted by [grunes](#) on Thu, 09 Feb 1995 18:49:41 GMT

[View Forum Message](#) <> [Reply to Message](#)

Archive-name: diagrami

Submitted-by: grunes@nrlvax.nrl.navy.mil (Mitchell R Grunes)

****Simple Diagramming Program for IDL & PV-Wave Language Programs****

This is an update.

Sorry, little or no help can be provided for this program.

-----CUT HERE-----

program diagrami !Diagrammer for IDL and PV-Wave

c EXAMPLE OF OUTPUT:

```
c +-----pro Sample,a,b,c          | 1
c |      a=indgen(15)^2              | 2
c |+-----if a eq b then begin    | 3
c ||      print,'A equals B'        | 4
c ||      c=0                        | 5
c |+-----else begin              | 6
c ||      print,'A does not equal B'| 7
c ||      c=1                        | 8
c |+-----endif                  | 9
c +-----end                      | 10
```

c Diagrams IDL and PV-Wave begin(or case)-end constructs, functions
c and procedures, places a * next to goto and return statements.

c

c Program by Mitchell R Grunes, ATSC/NRL. Revision date: 2/9/95.

c This program was written in FORTRAN, the One True Language.

c Note--this is a quick and dirty attempt--may not always work quite right.

c It does not yet handle CASE instances, since I don't use them myself,

c but it should draw a line around the entire CASE block.

c It can also be confused if an INCLUDE block contains a structure that

c begins inside and ends outside (or vice-versa).

c It is assumed that no fortran carriage control exists on the output

c file, so don't specify output to the screen or a terminal.

c I hope this works for you, but bear in mind that nothing short of

c a full-fledged language parser could really do the job. Perhaps

c worth about what you paid for it. (-:


```

elseif(j.le.160)then
  b(j:j)=a(i:i)
  j=j+1
endif
enddo
i=1
j=1
a=' '
iquote=0      !no ' yet
idquote=0     !no " yet
j=1
do i=1,160
  c=b(i:i)
  if(c.ge.'A'.and.c.le.'Z')c=char(ichar(c)+32)
  if(c.eq.";")goto 15
  if(c.eq."")iquote=1-iquote
  if(c.eq."")idquote=1-idquote
  if(iquote.eq.1)then
    if(find(a,'include ',2))then
      iquote=0
      idquote=0
    endif
  endif
  if(iquote.ne.0.or.idquote.ne.0)c=' '
  if(j.gt.1)then      !(kill multiple spaces)
    if(c.eq.' '.and.a(j-1:j-1).eq.' ')j=j-1
  endif
  a(j:j)=c
  j=j+1
enddo
15  i2=i1
    i3=i1
    iflag=0      !no goto on line
    if(find(a,'goto',8+32).or.find(a,'return',1+128))iflag=1
    if(find(a,'include ',2))then
      a=a(icol:160)
      if(.not.find(a,"",0))goto 16
      a=a(icol:160)
      if(.not.find(a,"",0))goto 16
      a(icol-1:160)=' '
      close(3)
      open(3,file=a,status='old')
      iunit=3
      nlinesave=nline
      i2=i2+1
      i3=i3+1
      i4=i4+1
16  continue

```

```

elseif(find(a,'endif ',2).or.find(a,'endfor ',2)
& .or.find(a,'endelse ',2).or.find(a,'endwhile ',2)
& .or.find(a,'endcase ',2))then
  i3=i3-1
  i4=i4+1
  if(find(a,'begin ',1))then
    i2=i2+1
    i3=i3+1
  endif
elseif(find(a,'case ',1))then
  i4=i4+1
  i2=i2+1
  i3=i3+1
elseif(find(a,'begin ',1))then
  i4=i4+1
  i2=i2+1
  i3=i3+1
elseif(find(a,'function ',2).or.find(a,'pro ',2))then
  if(lnSub.eq.0)then
    lnSub=1
    i2=i2+1
    i3=i3+1
    i4=i4+1
    if(i3.ne.1)then
      PRINT*, '***ERROR--INVALID DIAGRAMMING INDEX line',nline
      WRITE(2,*) '***ERROR--INVALID DIAGRAMMING INDEX!***'
      print*,char(7)
    endif
  endif
  i3=1
elseif(find(a,'end ',2))then
  if(i3.gt.0.or.lnSub.gt.0)then      !Problem: IDL end may actually
    i3=i3-1                        ! be an endif, endelse, etc.
    i4=i4+1
    if(i3.eq.0.and.lnSub.ne.0)lnSub=0
    if(find(a,'begin ',1))then
      i2=i2+1
      i3=i3+1
    endif
  else
    nMainEnd=nMainEnd+1
    print*, '***MAINLINE END line ',nline
    if(nMainEnd.gt.1)then
      PRINT*, '***ERROR--TOO MANY MAINLINE ENDS!***'
      WRITE(2,*) '***ERROR--TOO MANY MAINLINE ENDS!***'
      print*,char(7)
    endif
  endif
endif
endif

```

```

endif
icontinue=0
if(find(a,$ ',0))icontinue=1
a=' '
if(i1.lt.0.or.i2.lt.0.or.i3.lt.0.or.i4.lt.0)then
  PRINT*, '***ERROR--INVALID DIAGRAMMING INDEX line',nline
  WRITE(2,*)'***ERROR--INVALID DIAGRAMMING INDEX!***'
  print*,char(7)
  i1=max(i1,0)
  i2=max(i2,0)
  i3=max(i3,0)
  i4=max(i4,0)
endif
jj=max(1,min(16,2*i2-1))
if(i2.gt.0)a=aa(1:jj)
if(i4.ne.0)then
  jjj=1
  dowhile(jjj.lt.160.and.b(jjj:jjj).eq.' ')
    jjj=jjj+1
  enddo
  if(jjj.gt.1)b(1:jjj-1)=
& '-----'
  a(jj:16)='-----'
endif
do i=0,i4-1
  a(max(1,min(15,jj-i*2)):max(1,min(15,jj+1-i*2)))='+-'
enddo
i4=0
if(iline.ne.0.and.b(max(1,iline):160).eq.' ')then
  write(form,'(i5)')nline          !line #
  if(nline.eq.0)write(form,'(i5)')nlinesave
  if(form(1:1).eq.' ')form(1:1)='|'
  b(iline:iline+4)=form
endif
n=160
dowhile(n.gt.1.and.b(n:n).eq.' ')
  n=n-1
enddo
if(iflag.ne.0)a(1:1)='*'
c last minute change to reduce spaces in diagram block:
write(2,2)(a(i:i),i=1,15,2),(b(i:i),i=1,n)
2   format(132a1)
i1=i3
goto 10
99  if(iunit.eq.3)then
    iunit=1
    nline=nlinesave
    i1=i1-1

```

```

close(3)
goto 10
endif
end

```

C-----

logical function find(a,b,icond) !find b in a, subject to conditions:

!icond=sum of the following:

!1: Prior, if exists, must be blank

!2: Must be first non-blank

!4: Prior character, if present,

! must not be alphanumeric.

!8: Prior character, if present,

! must be blank or)

!16: Prior character, if present,

! must be blank or ,

!32: Next character not alphanumeric

!64: Next character not alphabetic

!128:Next character must be blank or (

character*(*) a,b

character*1 c,cNext

common icol

logical result

ii=len(a)

jj=len(b)

do i=1,ii-jj+1

if(a(i:i+jj-1).eq.b)then

icol1=i ! icol1=column of item found

icol =i+jj ! icol =colomn after item found

goto 10

endif

enddo

find=.false.

return

10 result=.true.

c=' '

cNext=' '

if(icol1.gt.1)c=a(icol1-1:icol1-1)

if(icol .le.ii)cNext=a(icol:icol)

if(result.and.iand(icond,1).ne.0.and.icol1.gt.1)then

result=c.eq.' '

endif

if(result.and.iand(icond,2).ne.0.and.icol1.gt.1)then

result=a(1:icol1-1).eq.' '

endif

```
if(result.and.iand(icond,4).ne.0)
& result=(c.lt.'0'.or.c.gt.'9').and.(c.lt.'a'.or.c.gt.'z')

if(result.and.iand(icond,8).ne.0)result=c.eq.' '.or.c.eq.'],'

if(result.and.iand(icond,16).ne.0)result=c.eq.' '.or.c.eq.'],'

if(result.and.iand(icond,32).ne.0)
& result=(cNext.lt.'0'.or.cNext.gt.'9').and.
&      (cNext.lt.'a'.or.cNext.gt.'z')

if(result.and.iand(icond,64).ne.0)
& result=(cNext.lt.'a'.or.cNext.gt.'z')

if(result.and.iand(icond,128).ne.0)
& result=cNext.eq.' '.or.cNext.eq.>('

find=result
end
```

-----CUT HERE-----

(opinions expressed are mine alone)
Mitchell R Grunes (grunes@nrlvax.nrl.navy.mil)
Allied-Signal Tech. Serv. / Naval Research Lab
