Subject: Re: When Ptr_New doesn't work
Posted by Malcolm Walters on Tue, 22 Jan 2002 14:51:46 GMT
View Forum Message <> Reply to Message

"Carles Badenes" <badenes@ieec.fcr.es> wrote in message
news:d46481f7.0201220342.3af528fa@posting.google.com...
> I have the following problem:
>
> rCoefficients=PtrArr(nElems)
> FOR i=0, nElems-1 DO BEGIN
> rCoefficients[i]=Ptr_New(PtrArr(elems[i]))
> ENDFOR
>
> since elems is a vector of integers, each element i of
> rCoefficients is a pointer to an array of elems[i] pointers.
> But, for some reason,
>
> ((*rCoefficients[j])[k]) = Ptr_New(FltArr(2), /NO_COPY)
>
> doesn't work. k and j are within the allowed range, of course, and
> ((*rCoefficients[j])[k]) is a null pointer, as expected from the
> initialization above. Ptr_New is supposed to allocate memory for the
> specified pointer to store an array of 2 floats, but I get the message
>
> Expression must be named variable in this context: <POINTER
> (<NullPointer>)>.
>
> I must be doing something wrong. Can you help?
>
> Thanks,
> Carles

"Carles Badenes" <badenes@ieec.fcr.es> wrote in message
news:<d46481f7.0201220342.3af528fa@posting.google.com>...
> I have the following problem:
>
> rCoefficients=PtrArr(nElems)
> FOR i=0, nElems-1 DO BEGIN
> rCoefficients[i]=Ptr_New(PtrArr(elems[i]))
> ENDFOR
>
> since elems is a vector of integers, each element i of
> rCoefficients is a pointer to an array of elems[i] pointers.
> But, for some reason,
>
> ((*rCoefficients[j])[k]) = Ptr_New(FltArr(2), /NO_COPY)
>
> doesn't work. k and j are within the allowed range, of course, and

> ((*rCoefficients[j])[k]) is a null pointer, as expected from the
> initialization above. Ptr_New is supposed to allocate memory for the
> specified pointer to store an array of 2 floats, but I get the message
>
> Expression must be named variable in this context: <POINTER
> (<NullPointer>)>.
>
> I must be doing something wrong. Can you help?
>
This seems to be due to how IDL allocates and dereferences its pointers, consider the code below. The first part seems to be what you are trying to do, I have just expanded it.
This doesn't work since when you do the inner 'Ptr_new' 'tempVar' moves in memory. I guess the error was that this change could not occur in you condensed ((*rCoefficients[j])[k]) command.
The solution is to create the inner part and then set the pointer to it afterwards.

I hope this is of help
Malcolm Walters

```
PRO TEST

nElems=2
elems=[3,2,1]

rCoefficients=PtrArr(nElems)
FOR i=0, nElems-1 DO BEGIN
   rCoefficients[i]=Ptr_New(PtrArr(elems[i]))
   tempVar=(*rCoefficients[i])
   FOR j=0, elems[i]-1 DO BEGIN
    tempVar[j] = Ptr_New(FltArr(2), /NO_COPY)
    ; tempVar no longer equals *rCoefficients[i]
    (*tempVar[j])[0] = 100-i
    (*tempVar[j])[1] = 200-j
   ENDFOR
ENDFOR

rCoefficients=PtrArr(nElems)
FOR i=0, nElems-1 DO BEGIN
   tempVar=(PtrArr(elems[i]))
   FOR j=0, elems[i]-1 DO BEGIN
    tempVar[j] = Ptr_New(FltArr(2), /NO_COPY)
    ; tempVar no longer equals *rCoefficients[i]
    (*tempVar[j])[0] = 100-i
    (*tempVar[j])[1] = 200-j
   ENDFOR
```

```
    rCoefficients[i]=Ptr_New(tempVar)
ENDFOR


FOR i=0, nElems-1 DO BEGIN
 FOR j=0, elems[i]-1 DO BEGIN
  print,i,j,(*(*rCoefficients[i])[j])
 ENDFOR
ENDFOR

END
```