

Re: A basic question about linked list

Source: <http://coding.derkeiler.com/Archive/Fortran/comp.lang.fortran/2007-07/msg00898.html>

- *From:* glen herrmannsfeldt <gah@xxxxxxxxxxxxxxxxxxx>
 - *Date:* Wed, 18 Jul 2007 16:56:01 -0800
-

veda wrote:

I want to know the correct procedure for deallocating memory that gets allocated during creation of a linked list (I am using CVF).

(snip)

```
type curve
type(face_1d):: facet
type(curve), pointer::next
end type curve
```

```
type(curve), pointer:: head
type(curve), pointer::tail
type(curve), pointer::ptr_temp
```

(snip)

Later in the code, I want to re-define the curve identified by "head" (e.g. all together different set of segments). What do I need to do to deallocate memory correctly for the previously defined curve before starting to define the new curve?

If is usual to pointer assign NULL() to the next pointer of the last element in the list, which usually means you don't need a tail pointer. In any case, you go down the list, saving the next pointer in a temporary pointer, deallocating the current element, and then copying the temporary to the current pointer.

Maybe something like:

Re: A basic question about linked list

```
do while(associated(head))
ptr_temp => head%next
deallocate(head)
head => ptr_temp
enddo
head => null()
```

If the list isn't terminated with a null pointer, you could use

```
do while(.not.associated(head,tail))
```

but null termination is better.

-- glen

.