Subject: Testing for structure equality during loops
Posted by evan.jones on Wed, 27 May 2015 15:48:07 GMT
View Forum Message <> Reply to Message

Good morning everyone. I have some structures within a class that I want to loop through. Take the following class as an example and pretend that I have included the init, getProperty, and setProperty methods - and that the structure field values in example_class__define are those assigned in the init method.

```
example_class::method1
  for *self.var3 = self.var1, self.var2 do begin
   print, 99
   *self.var3 = self.var2
  endfor
end

example_class::method2
  while self.var1 NE self.var2 do begin
   print,99
   self.var1 = self.var2
  endwhile
end

example_class__define
struct:{name,$
   var1:{name,a:1,b:2,c:ptr_new(1)},$
   var2:{name,a:1,b:2,c:ptr_new(10)},$
   var3:ptr_new(/allocate_heap)}
end
```

Method1 won't compile and when I try to run method22 I receive this sort of error: "Struct expression not allowed in this context: <PtrHeapVar115>." My program is a bit more complicated as I am trying to loop through linked lists, but hopefully these simplified examples convey my issue. How might I accomplish the task of looping through structures by testing for the equality or 'sameness' of all the fields of different structures without testing the equality of each corresponding field individually?