

---

Subject: Re: Bug in SWITCH - ELSE: statement?

Posted by [Paul Van Delst\[1\]](#) on Thu, 11 Oct 2007 16:27:44 GMT

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

---

Jean H wrote:

```
>> You brought up something interesting (to me anyway.) If you place
>> the
>> ELSE earlier in the SWITCH statement, then everything below it is
>> executed. If you do the same for a CASE statement then only then
>> ELSE is executed. It never occurred to me until I saw your post
>> that it was possible to place the ELSE anywhere but at the end.
>
> This is useless and can bring serious headaches...
>
> Both the Case and Switch run from the 1st condition to the last one...
> so anything you write below the "else" will never (Case) or always
> (Switch) be executed... then there is no need for a Case or Switch
> anymore! ... just delete (Case) your code or write it outside of the
> Switch block!
```

Goodness.... I never even thought of that. Why on Earth should the order of the case selectors matter in a CASE construct? I just wrote:

```
pro test_case, number
  case number of
    else: print, 'I do not like this number!'
    1: print, '1 is a good number'
    2: print, '2 is even better'
    3: print, '3 is weird looking'
  endcase
end
```

```
IDL> .run test_case
% Compiled module: TEST_CASE.
IDL> test_case, 1
I do not like this number!
IDL> test_case, 3
I do not like this number!
IDL> test_case, 2
I do not like this number!
IDL> test_case, 1000
I do not like this number!
```

That is just ridiculous behaviour.

The Fortran95 equivalent:

```
program test_case
  integer :: number
  write(*, /, "Enter a number: ") , advance='no'
  read(*, '(i10)') number

  select case (number)
  case default
    write(*, *) 'I do not like this number!'
  case (1:3)
    write(*, *) number, ' is a good number!'
  end select

end program test_case
```

doesn't depend on the order of the "CASE"s:

```
lnx:scratch : lf95 --f95 test_case.f90
Encountered 0 errors, 0 warnings in file test_case.f90.
lnx:scratch : a.out
```

```
Enter a number:3
3 is a good number!
lnx:scratch : a.out
```

```
Enter a number:1000
I do not like this number!
lnx:scratch : a.out
```

```
Enter a number:2
2 is a good number!
```

that is what I would expect to happen. Seems like the IDL case construct cribbs code from the switch construct. Tsk tsk.

cheers,

paulv

---