Derived Types in Fortran

Victor Eijkhout, Harika Gurram,
Je'aime Powell, Charley Dey

Fall 2018

Structures

Structures: `type`

The Fortran name for structures is `type` or *derived type*.

Type definition

`Type name` / `End Type` block. Variable declarations inside the block

```
type mytype
  integer :: number
  character :: name
  real(4) :: value
end type mytype
```

Creating a type structure

Declare a typed object in the main program:

```
Type(mytype) :: typed_object,object2
```

Initialize with type name:

```
typed_object = mytype( 1, 'my_name', 3.7 )
object2 = typed_object
```

Member access

Access structure members with `%`

```
Type(mytype) :: typed_object
type_object%member = ....
```

Example

```
type point
   real :: x,y
end type point

type(point) :: p1,p2
p1 = point(2.5, 3.7)

p2 = p1
print *,p2%x,p2%y
```

Types as subprogram argument

```
real(4) function length(p)
  implicit none
  type(point),intent(in) :: p
  length = sqrt( p%x**2 + p%y )
end function length
```