

```

1  !-----
2  ! Variables
3  !-----
4
5  ! Scalars
6  integer :: n
7  integer(kind=selected_int_kind(10)) :: large_integer
8
9  real :: pi, diameter
10 character(len=30) :: text
11
12 pi = 3.141592
13
14 ! Arrays
15 real, dimension(3) :: point
16 integer, dimension(4,4) :: matrix
17
18 point(1) = 1.23456
19 point = (/ 1.23, 4.56, 7.89 /)
20 matrix(1,2) = 4
21
22 !-----
23 ! Conditionals
24 !-----
25
26 if( diameter < 0 ) then
27     ! ...
28 else if( diameter == 0 ) then
29     ! ...
30 else
31     ! ...
32 end if
33
34 select case( n )
35     case( 1 )
36         ! ...
37     case( 2:3 )
38         ! ...
39     case default
40         ! ...
41 end select
42
43 !-----
44 ! Program units
45 !-----
46
47 ! Main program
48 program my_prog
49     use my_mod
50     implicit none
51     ! ...
52 contains
53     subroutine my_sub( arg, gra )
54         integer, intent(in out) :: arg
55         real, intent(out) :: gra
56         ! ...
57     end subroutine my_sub
58 end program my_prog
59
60 ! Module
61 module my_mod
62     implicit none
63     integer, parameter :: bar
64 contains
65     function my_fun( arg ) result( foo )
66         real, intent(in) :: arg
67         integer :: foo
68         ! ...
69     end function my_fun
70 end module my_mod
71
72 !-----
73 ! Input/output
74 !-----
75
76 read *, diameter
77 read(unit=*,fmt=*) diameter
78
79 print *, "Diameter = ", diameter
80 write(unit=*,fmt=*) "pi = ", pi
81

```