Fortran 95

Dobre złego początki
czyli
stałe, zmienne i operatory

Waldemar Kulig
Zakład Chemii Teoretycznej
UJ

Plan

- Komentarze
- Zmienne i stałe
- Operatory

Komentarze

```
integer i ! to jest komentarz
! integer i  to też jest komentarz
```

- Dlaczego warto pisać komentarze???

... bo po kilku latach raczej trudno jest odgadnąć jak dany program działa szczególnie jeżeli zawiera kilka tysięcy linii kodu

Zmienne

Zmienne – nazwy (wybrane przez programistę), które unikalnie identyfikują obiekt (informację) przechowywaną w pamięci komputera.

Przykład:

w programie pewne wartości (3.1415 ; 7 ; -12.5)
są nazywane jako (pi ; dzien_tygodnia, temperatura)

Wartości zmiennych mogą ulegać zmianie w trakcie wykonywania programu (np. temperatura), bądź mogą pozostawać niezmienione (np. pi)

Nazwy zmiennych

- mogą zawierać litery, cyfry i znak podkreślenia
- muszą zaczynać się od litery!!!
- nie mogą być dłuższe niż 31 znaków
- nie są czułe na wielkość liter

```
integer :: _bla, 1las ! niepoprawne
integer :: konstantynoneapolitanczykiewiczowna ! niepoprawne
real(8) :: a_mos, z321, objetosc  ! poprawne
```

```
real(8) :: name, Name, NAME, naMe
                                  ! poczwórna deklaracja tej samej zmiennej
```

Podziały zmiennych

1) ze względu na typ
   - całkowite
   - rzeczywiste
   - tekstowe
   - logiczne
   - zespolone

2) ze względu na sposób deklaracji
   - jawnie deklarowane (*implicit none*)
   - niejawnie deklarowane

3) ze względu na strukturę
   - proste
   - indeksowane (tablice)

Podziały zmiennych

4) ze względu na ilość zajmowanej pamięci
   - 1, 2, 4, 6, 8 i 10 bajtowe
   - o dowolnej liczbie bajtów

5) ze względu na rodzaj przydzielanej pamięci
   - statyczne
   - dynamiczne

Podstawowe typy zmiennych

1) typ całkowity - *integer*

integer
integer([KIND=]kind-parameter) KIND = 1,2,4,8

```
integer :: tralala ! zmienna całkowita zajmująca 4 bajty
integer(KIND=8) :: tralala ! zmienna całkowita zajmująca 8 bajtów
```

2) typ rzeczywisty - *real*

real
real([KIND=]kind-parameter) KIND = 4,8,16

```
real :: blabla ! zmienna rzeczywista zajmująca 4 bajty
real(KIND=8) :: blabla ! zmienna rzeczywista zajmująca 8 bajtów
```

Podstawowe typy zmiennych

3) typ znakowy - *character*

character
character([LEN=]length)

KIND = 1

```
character :: c ! zmienna znakowa zawsze jednobajtowa!
character(10) :: c ! zmienna znakowa o długości 10 znaków
character(len=12) :: c ! zmienna znakowa o długości 12 znaków
```

4) typ logiczny - *logical*

logical
logical([KIND=]kind-parameter)

KIND = 1,2,4,8

```
logical :: x ! zmienna logiczna czterobajtowa
logical(KIND=8) :: x ! zmienna logiczna zajmująca 8 bajtów
```

Stałe

```
-255 ! stała całkowita czterobajtowa w zapisie dziesiętnym

-1.234 ! stała zmiennoprzecinkowa czterobajtowa
1.234e2 ! stała zmiennoprzecinkowa czterobajtowa
1.234d2 ! stała zmiennoprzecinkowa ośmiobajtowa

.TRUE. ! stała logiczna

''a@#sdfsf13'' ! stała znakowa
'123567' ! stała znakowa
'ala' ! stała znakowa
```

Deklarowanie zmiennych

Deklarowanie zmiennych w programie:

```
program deklaracja
implicit none

 ! blok deklaracji zmiennych
real:: masa !deklaracja zmiennej rzeczywistej o nazwie masa
integer:: ilosc_ksiazek ! deklaracja zmiennej całkowitej o nazwie
                                    ! ilosc_ksiazek
.....

! koniec bloku deklaracji

! blok instrukcji
! koniec bloku instrukcji

end
```

Inicjalizowanie zmiennych

..... czyli nadawanie im wartości początkowych

```
program deklaracja2
implicit none

 ! blok deklaracji zmiennych
real:: masa !deklaracja zmiennej rzeczywistej o nazwie masa
integer:: ilosc_ksiazek ! deklaracja zmiennej całkowitej o nazwie
                                    ! ilosc_ksiazek
! koniec bloku deklaracji

! blok instrukcji
masa = 12.5
ilosc_ksiazek = 127
! koniec bloku instrukcji

end
```

Operatory

1) Arytmetyczne

+, -, *, /
** ! potęgowanie
operatory są lewostronnie łączne ???

2) Relacji

> lub .gt. ! większe od
< lub .lt. ! mniejsze od
>= lub .ge. ! większe lub równe
<= lub .le. ! mniejsze lub równe
== lub .eq. ! równe
/= lub .ne. ! różne