

Programowanie w języku Fortran

dr inż. Maciej Woźniak ¹

¹Katedra Informatyki, Wydział Informatyki, Elektroniki i Telekomunikacji,
Akademia Górniczo-Hutnicza, Kraków, Polska

Standardowa definicja string

```
character(7) :: string1  
character(len=7) :: string2
```

String o zmiennej długości

```
character(len=:), allocatable :: string3  
string3 = "Text"
```

String o znanej wartości, ale bez podawania długości

```
character(len=*), parameter :: text = "Hello World"
```

Długość string sprawdzamy za pomocą **len**

```
write(*,*) len(" ") ! 1  
write(*,*) len("test") ! 4
```

Przypisywanie do zbyt krótkich string zakończy się ucięciem tekstu

```
character(3) :: name  
name = "Janek"  
write(*,*) name ! Jan
```

Substring

Do fragmentów string dostajemy się podobnie jak do podtablic.
Możemy pominąć jedną z wartości wewnątrz nawiasów.

```
character(9) :: text  
text = "substring"  
write(*,*) text(1:3) ! sub  
write(*,*) text(4:9) ! string  
write(*,*) text(:3) ! sub  
write(*,*) text(4:) ! string
```

W podobny sposób można podmieniać kawałki tekstu

```
character(9) :: text  
text = "substring"  
text(1:3) = "max" ! sub  
write(*,*) text ! maxstring
```

Pozycję podstring można znaleźć za pomocą **index**.

```
write(*,*) index("banana", "nana") ! 3  
write(*,*) ("banana", "test") ! 0
```

Do konkatencji string służy //

```
write(*,*) "ba" // "nana" ! banana  
write(*,*) "ba", "nana" ! ba nana
```

Do sprawdzenia kodu ASCII litery służy funkcja **iachar**

```
write(*,*) iachar("A") ! 65
```

Do sprawdzenia litery odpowiadającej kodowi ASCII służy funkcja **achar**

```
write(*,*) achar(65) ! A
```

integer, target :: t

integer, target :: ta(10)

integer, pointer :: p

integer, pointer, dimension(:) :: pa

! p staje sie wskaźnikiem na skalar

p => t

! pa staje sie wskaźnikiem na tablicę

pa => ta

! pa staje się null-em

pa => null()

Wskaźniki na funkcje

Wskaźnik na subroutinę o interfejsie mysub

```
procedure (mysub), pointer :: mpsub  
...  
call mpsub()
```

Tablica wskaźników na subroutinę o interfejsie mysub

```
type :: sub_type  
  procedure (mysub), pointer, nopass :: mpsub  
  ! nopass wskazuje na to, że pierwszy argument nie jest this  
end type  
type (sub_type) :: app(10)  
...  
app(1) => some_procedure  
...  
call app(1)%mpsuh()
```