

The Fortran “type” command

Σύνθετοι τύποι δεδομένων στη Fortran:
μια σύντομη εισαγωγή

Μάκης Χουρδάκης
23/5/2012

Under Creative Commons Attribution Share-Alike License

Σύνταξη της εντολής type

Program TheGrade

implicit none

```
type student
  character*15 :: name
  integer      :: id, grade
end type student
```

```
type(student) :: you
```

```
Write (*,*) "What is your name?"
Read (*,*) you%name
```

```
Write (*,*) "What is your grade?"
Read (*,*) you%grade
```

```
you%grade = you%grade + 1
Write (*,*) "Get +1! Your grade now is: ", you%grade
```

End

Ορισμός του τύπου
"student" πριν τη
δήλωση μεταβλητών

Ορισμός της μεταβλητής
"you" ως τύπου "student"

Ανάγνωση του πεδίου
"grade" της μεταβλητής "you"

*Η μεταβλητή "you" τελικά είναι ένας σύνθετος
τύπος με πολλά διαφορετικά πεδία!*

Διάνυσμα: ως array

Program Vector

implicit none

Real :: A(4)

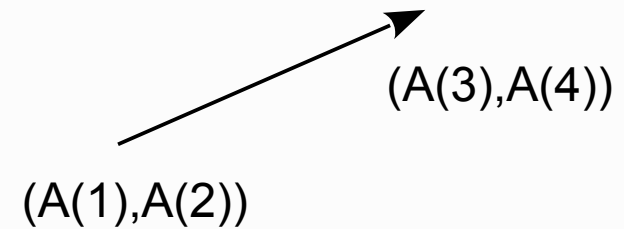
A(1) = 1 ; A(2) = 1 ! (x1,y1)=(1,1)

A(3) = 2 ; A(4) = 4 ! (x2,y2)=(2,4)

Write (*,*) "A start = (" , A(1), ", " , A(2), ")"

Write (*,*) "A end = (" , A(3), ", " , A(4), ")"

End



Διάνυσμα: ως τύπος (1)

Program Vector2

implicit none

Type vector

Real :: x1, y1, x2, y2

end type

type(vector) :: A

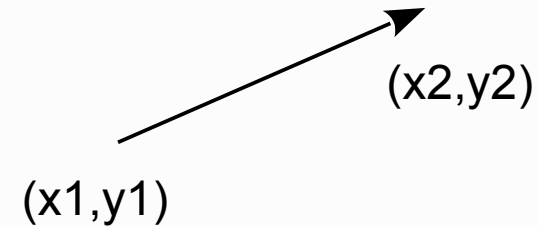
A%x1 = 1 ; A%y1 = 1 ! (x1,y1)=(1,1)

A%x2 = 2 ; A%y2 = 4 ! (x2,y2)=(2,4)

Write (*,*) "A start = (" , A%x1, ", " , A%y1, ")"

Write (*,*) "A end = (" , A%x2, ", " , A%y2, ")"

End



Διάνυσμα: ως τύπος (2)

Program Vector3

implicit none

Type point
Real :: x,y
end type

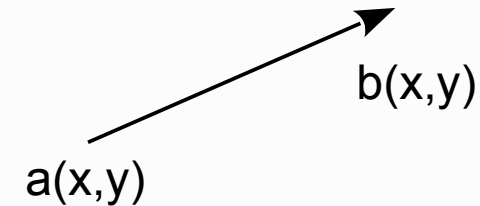
Type vector
Type(point) :: a,b
end type

type(vector) :: A

A%a%x = 1 ; A%a%y = 1 ! (x1,y1)=(1,1)
A%b%x = 2 ; A%b%y = 4 ! (x2,y2)=(2,4)

Write (*,*) "A start = (" , A%a%x, ", " , A%a%y, ")"
Write (*,*) "A end = (" , A%b%x, ", " , A%b%y, ")"

End



Ο τύπος "vector"
χρησιμοποιεί
τον τύπο "point"

Διάνυσμα: ως τύπος (3)

Program Vector3

implicit none

```
Type point  
  Real :: x,y  
end type
```

```
Type vector  
  Type(point) :: a,b  
end type
```

```
type(point) :: p,q  
type(vector) :: A
```

```
p%x = 1 ; p%y = 1    ! (x1,y1)=(1,1)
```

```
q%x = 2 ; q%y = 4    ! (x2,y2)=(2,4)
```

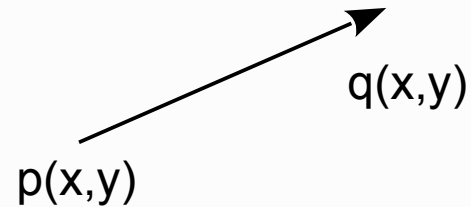
```
A%a = p
```

```
A%b = q
```

```
Write (*,*) "A start = (", A%a%x, ", ", A%a%y, ")"
```

```
Write (*,*) "A end = (", A%b%x, ", ", A%b%y, ")"
```

End



Ορίζω τις συντεταγμένες των δυο σημείων

Ορίζω την αρχή και το πέρας του διανύσματος

Διάνυσμα: ως τύπος (4)

Program Vector3

implicit none

```
Type point
  Real :: x,y
end type
```

```
Type vector
  Type(point) :: a,b
end type
```

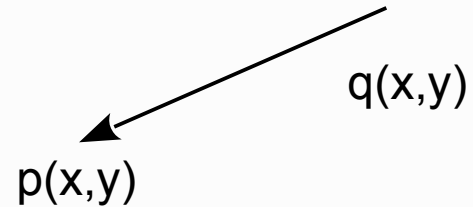
```
type(point) :: p,q,temp
type(vector) :: A
```

```
p%x = 1 ; p%y = 1   ! (x1,y1)=(1,1)
q%x = 2 ; q%y = 4   ! (x2,y2)=(2,4)
```

```
temp = p
p = q
q = temp
```

```
A%a = p
A%b = q
```

End



Αντιμεταθέτω ολόκληρα τα
“αντικείμενα” σαν να ήταν
απλές μεταβλητές

Arrays σύνθετων τύπων

Program ObjectsArray

implicit none

Type point

Real :: x,y
end type

integer :: i
type(point) :: a(3)

a%x = 0 ; a%y=0

Do i=1,3

a(i)%x = i
a(i)%y = 2*i
Write (*,*) a(i)%x, a(i)%y
enddo

End

Όπως θα δήλωνα μια
array πραγματικού τύπου:
Real :: b(3)

Μηδενίζω τα x,y όλων
των αντικειμένων
(αντίστοιχα σε array: b=0)

Arrays σύνθετων τύπων (2)

Program ObjectsArray2

implicit none

Type point

Real :: x,y

end type

integer :: i,j

type(point), allocatable :: a(: , :)

a%x = 0 ; a%y=0

allocate(a(3,3))

...

End

Ακριβώς όπως σε απλές arrays!