

## Άσκηση 1

**A.** Γράψτε τις αντίστοιχες εντολές FORTRAN για τον υπολογισμό της τιμής του  $y$  σύμφωνα με τη σχέσεις:

$y = \frac{5x^2 - \sqrt{x}}{x-1}$	$y = \frac{x^2}{2} - 1 + \frac{\sqrt{1}}{x-1}$	$y = \sqrt{1+x^3} \frac{4}{x+1}$	$y = x^{-4} + 5x + \frac{1}{x-1} + 2$
-----------------------------------	--	----------------------------------	---------------------------------------

```
y = (5*x**2 - sqrt(x))/(x - 1)
```

```
y = x**2/2 - 1 + sqrt(1.)/(x - 1)
```

```
y = sqrt(1 + x**3)*4/(x + 1)
```

```
y = x**(-4) + 5*x + 1/(x -1) + 2
```

**B.** Να γραφεί κώδικας FORTRAN που θα δέχεται από το χρήστη μια πραγματική τιμή  $x$ . Στη συνέχεια θα τυπώνει τις τιμές του  $y$  για κάθε μια από τις παραπάνω σχέσεις.

```
program exercise2
IMPLICIT NONE
real x, y

print *, 'Give a real value'
read *, x

y = (5*x**2 - sqrt(x))/(x - 1)
print *, 'Value of y =',y

y = x**2/2 - 1 + sqrt(1.)/(x - 1)
print *, 'Value of y =',y

y = sqrt(1 + x**3)*4/(x + 1)
print *, 'Value of y =',y

y = x**(-4) + 5*x + 1/(x -1) + 2
print *, 'Value of y =',y

end
```

## Άσκηση 2

Να γραφεί κώδικας FORTRAN που θα δέχεται από το χρήστη μια πραγματική τιμή  $x$ . Στη συνέχεια θα τυπώνει την τιμή του  $y$  σύμφωνα με τις σχέσεις:

$$y = \frac{5}{x+2} \text{ για } x \geq 1$$

$$y = \sqrt{|x|+1} \text{ για } x < 1$$

**ΣΗΜΕΙΩΣΗ:** Είναι απαραίτητη η χρήση δομής IF. Να γράψετε και τις τρεις παραλλαγές του παραπάνω κώδικα που θα χρησιμοποιούν αντίστοιχα τη Δομή 1, τη Δομή 2 και τη Δομή 3.

### ΔΟΜΗ 1

```
program exercise3
  IMPLICIT NONE
  real x,y

  print *, 'Give a real value'
  read *, x

  if(x>=1.) then
    y=5./(x+2.)
  endif

  if(x<1.) then
    y=sqrt(abs(x)+1.)
  endif

  print *, 'Value of y =',y

end
```

### ΔΟΜΗ 2

```
program exercise3
  IMPLICIT NONE
  real x,y

  print *, 'Give a real value'
  read *, x

  if(x>=1.) then
```

```

    y=5./(x+2.)
else
    y=sqrt(abs(x)+1.)
endif

print *, 'Value of y =',y

end

```

### ΔΟΜΗ 3

```

program exercise3
IMPLICIT NONE
real x,y

print *, 'Give a real value'
read *, x

if(x>=1.) y=5./(x+2.)

if(x<1.) y=sqrt(abs(x)+1.)

print *, 'Value of y =',y

end

```

## Άσκηση 3

Να γραφεί κώδικας FORTRAN που θα δέχεται από το χρήστη μια πραγματική τιμή  $x$ . Στη συνέχεια θα τυπώνει την τιμή της συνάρτησης  $f(x)$ :

$$f(x) = \frac{5\sqrt{x}}{x-2} + \frac{3}{5}$$

Ο κώδικας θα πρέπει να τυπώνει μήνυμα σφάλματος, αν ο χρήστης δώσει τιμή του  $x$  στην οποία δεν ορίζεται η παραπάνω συνάρτηση.

```

program exercise4
IMPLICIT NONE
real x,y

print *, 'Give a real value'
read *, x

if((x<0.).OR.(x==2)) then
    print *, 'Error'

```

```
else
  y=5*sqrt(x)/(x-2)+3./5
  print *, 'Value of y =',y
endif

end
```

## Άσκηση 4

Να γραφτεί πρόγραμμα σε FORTRAN που να τυπώνει τους η πρώτους φυσικούς αριθμούς για τους οποίους το άθροισμα των τετραγώνων τους δεν ξεπερνά το 200. Να γράψετε **δύο εκδοχές**:

α) με συνδυασμένη χρήση do με μετρητή και if

```
program main
IMPLICIT NONE
integer i, s

s=0
do i=1,1000
  s=s+i**2
  if(s>200) exit
  print *, i
enddo

end
```

β) με χρήση αέναης επανάληψης (do χωρίς μετρητή).

```
program main
IMPLICIT NONE
integer i, s

i=0
s=0
do
  i=i+1
  s=s+i**2
  if(s>200) exit
  print *, i
enddo

end
```

## Άσκηση 5

Να υπολογιστεί το παραγοντικό ενός θετικού ακέραιου αριθμού. Το παραγοντικό ενός θετικού ακέραιου αριθμού  $N \neq 0$  συμβολίζεται  $N!$  και ορίζεται ως το γινόμενο των αριθμών 1, 2, 3, . . . , (N-1), N. Αν  $N=0$  τότε  $N! = 1$ .

**Παράδειγμα:**  $5! = 1 \cdot 2 \cdot 3 \cdot 4 \cdot 5 = 120$

Ελέγξτε τα αποτελέσματα του κώδικά σας σύμφωνα με τον παρακάτω πίνακα (Πηγή: <http://en.wikipedia.org/wiki/Factorial>)

N	N!
0	1
1	1
2	2
3	6
4	24
5	120
6	720
7	5 040
8	40 320
9	362 880
10	3 628 800
15	1 307 674 368 000
20	2 432 902 008 176 640 000

Να γράψετε **δύο εκδοχές**:

α) Με χρήση integer μεταβλητών

```
program factorial_of_N
IMPLICIT NONE
integer N,i
integer factorial

print *, 'This program computes the factorial of &
      & a positive integer number N'
print *, 'Give the N'
read *, N

factorial=1
do i = 1, N
  factorial = factorial*i
enddo

print *, 'The factorial of N is'
print *, factorial
```

end

β) Με χρήση real μεταβλητών

```
program factorial_of_N
  IMPLICIT NONE
  integer N,i
  real factorial

  print *, 'This program computes the factorial of &
          & a positive integer number N'
  print *, 'Give the N'
  read *, N

  factorial=1.
  do i = 1, N
    factorial = factorial*i
  enddo

  print *, 'The factorial of N is'
  print *, factorial

end
```

## Άσκηση 6

Δίνεται η σειρά:

$$\sum_{i=1}^{\infty} \frac{1}{2^i} = \frac{1}{2} + \frac{1}{4} + \frac{1}{8} + \dots$$

Προεπιλέγοντας τον αριθμό των όρων υπολογίστε την τιμή της σειράς.  
Ελέγξτε τα αποτελέσματά σας σύμφωνα με τον παρακάτω πίνακα:

N	$\sum_{i=1}^{\infty} \frac{1}{2^i}$
1	0.5
2	0.75
3	0.875
4	0.9375
5	0.96875
10	0.9990234375

Υπάρχει κάποιο άνω όριο στις τιμές της σειράς αυτής;  
Συγκρίνετε τα αποτελέσματά ως προς τη χρήση single και double precision για την μεταβλητή της τιμής της σειράς.

```

program series
IMPLICIT NONE
integer N,i
real s

print *, 'Give the number of terms'
read *, N

s=0.
do i = 1, N
  s = s + 1./2.**i
enddo

print *, 'The result is'
print *, s

end

```

## Άσκηση 7

Να γραφεί κώδικας FORTRAN που θα υπολογίζει το άθροισμα όλων των στοιχείων ενός διανύσματος **a** (μονοδιάστατη array) διάστασης N. Τα στοιχεία του διανύσματος **a** δίνονται από τη σχέση:  $a(i) = 2^i$

Ο υπολογισμός να γίνει:

α) με την τεχνική αθροίσματος σε μεταβλητή

```

program vector
IMPLICIT NONE
integer, allocatable, dimension(:):: a
integer N, i, s

print *, 'Give the dimension of the vector'
read *, N

allocate(a(N))

do i = 1, N
  a(i) = 2*i
enddo

s = 0

do i = 1, N
  s = s + a(i)
enddo

print *, 'sum=', s

```

```
deallocate(a)
end
```

β) με τη χρήση της εσωτερικής συνάρτησης `sum( )`

```
program vector
IMPLICIT NONE
integer, allocatable, dimension(:):: a
integer N, i, s

print *, 'Give the dimension of the vector'
read *, N

allocate(a(N))

do i = 1, N
    a(i) = 2*i
enddo

s = sum(a)

print *, 'sum=', s

deallocate(a)
end
```

## Άσκηση 8

Να γραφεί κώδικας FORTRAN που θα υπολογίζει το άθροισμα όλων των στοιχείων ενός τετραγωνικού πίνακα **A** (διδιάστατη array) διάστασης NxN. Τα στοιχεία του πίνακα **A** δίνονται από τη σχέση:  $A(i, j) = 2*i - j$

Ο υπολογισμός να γίνει:

α) με την τεχνική αθροίσματος σε μεταβλητή

```
program matrix
IMPLICIT NONE
integer, allocatable, dimension(:,):: A
integer N, i, j, s

print *, 'Give the dimension of the square matrix'
read *, N

allocate(A(N,N))

do i = 1, N
    do j = 1, N
```



```
        A(i,j) = 2*i - j
    enddo
enddo
```

```
s = 0
```

```
do i = 1, N
    do j = 1, N
        s = s + A(i,j)
    enddo
enddo
```

```
print *, 'sum=', s
```

```
deallocate(A)
end
```

β) με τη χρήση της εσωτερικής συνάρτησης sum( )

```
program matrix
```

```
IMPLICIT NONE
```

```
integer, allocatable, dimension(:,:):: A
```

```
integer N, i, j, s
```

```
print *, 'Give the dimension of the square matrix'
```

```
read *, N
```

```
allocate(A(N,N))
```

```
do i = 1, N
    do j = 1, N
        A(i,j) = 2*i - j
    enddo
enddo
```

```
s = sum(A)
```

```
print *, 'sum=', s
```

```
deallocate(A)
end
```