



ΑΡΙΣΤΟΤΕΛΕΙΟ
ΠΑΝΕΠΙΣΤΗΜΙΟ
ΘΕΣΣΑΛΟΝΙΚΗΣ

ΑΝΟΙΧΤΑ
ΑΚΑΔΗΜΑΪΚΑ
ΜΑΘΗΜΑΤΑ



Εισαγωγή στον Προγραμματισμό Η/Υ (Fortran 90/95/2003)

Ενότητα 5: Εντολές Επανάληψης

Νίκος Καραμπετάκης
Τμήμα Μαθηματικών



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο

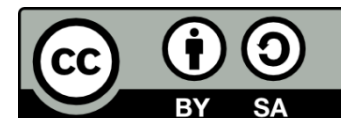


ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



ΕΣΠΑ
2007-2013
πρόγραμμα για την ανάπτυξη
ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ



Άδειες Χρήσης

- Το παρόν εκπαιδευτικό υλικό υπόκειται σε άδειες χρήσης Creative Commons.
- Για εκπαιδευτικό υλικό, όπως εικόνες, που υπόκειται σε άλλου τύπου άδειας χρήσης, η άδεια χρήσης αναφέρεται ρητώς.



Χρηματοδότηση

- Το παρόν εκπαιδευτικό υλικό έχει αναπτυχθεί στα πλαίσια του εκπαιδευτικού έργου του διδάσκοντα.
- Το έργο «Ανοικτά Ακαδημαϊκά Μαθήματα στο Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης» έχει χρηματοδοτήσει μόνο την αναδιαμόρφωση του εκπαιδευτικού υλικού.
- Το έργο υλοποιείται στο πλαίσιο του Επιχειρησιακού Προγράμματος «Εκπαίδευση και Δια Βίου Μάθηση» και συγχρηματοδοτείται από την Ευρωπαϊκή Ένωση (Ευρωπαϊκό Κοινωνικό Ταμείο) και από εθνικούς πόρους.



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



ΕΣΠΑ
2007-2013
πρόγραμμα για την ανάπτυξη
ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ



Περιεχόμενα Ενότητας

1. Η έννοια του μετρητή.
2. Ανακύκλωση και χρήση μετρητών σε προγράμματα.
3. Η εντολή DO.
4. Οι τρεις μορφές της DO-END DO.
5. Φωλιασμένες εντολές επανάληψης.



Σκοποί Ενότητας

1. Η ανάλυση της έννοιας ανακύκλωση.
2. Η παρουσίαση των μορφών της εντολής επανάληψης DO-END DO.
3. Η παρουσίαση του συνδυασμού των δύο ή παραπάνω εντολών επανάληψης DO-END DO (Φωλιασμένες εντολές επανάληψης).



Παραδείγματα που απαιτούν εντολές επανάληψης

Παράδειγμα 1. Έστω ότι μας ζητείται να βρούμε το άθροισμα 10 αριθμών. Τότε θα πρέπει να εκτελέσουμε 10 φορές τις εξής ενέργειες :

α) Δώσε έναν αριθμό,

β) Πρόσθεσε τον αριθμό αυτόν σε μια μεταβλητή έστω S, την οποία θα χρησιμοποιούμε για να αποθηκεύουμε το συνολικό άθροισμα των αριθμών.

Παράδειγμα 2. Έστω ότι θέλουμε να βρούμε το μέγιστο κοινό διαιρέτη (ΜΚΔ) δύο ακεραίων αριθμών α , β βάσει του τύπου

$$\text{ΜΚΔ}(\alpha, \beta) = \begin{cases} \text{ΜΚΔ}(\alpha - \beta, \beta) & \alpha > \beta \\ \text{ΜΚΔ}(\alpha, \beta - \alpha) & \alpha < \beta \end{cases}$$

Η παραπάνω διαδικασία θα πρέπει να επαναληφθεί ένα πλήθος φορών που δε γνωρίζουμε, έως ότου οι α , β να γίνουν ίσοι.



Ανακύκλωση και χρήση μετρητών σε προγράμματα

1. Η έννοια του μετρητή

Παράδειγμα. Να γραφεί πρόγραμμα που θα υπολογίζει το μέσο όρο N αριθμών.

```
S=0
```

```
10 READ*,X
```

```
S=S+X
```

```
GOTO 10
```

Εκτελείται άπειρες φορές ! Χρειάζεται κάποιος να μετράει τον αριθμό φορών επανάληψης.

Ο μετρητής : Μεταβλητή που αυξάνει κατά ένα κάθε φορά που εκτελείται μια επανάληψη.



Ανακύκλωση και χρήση μετρητών σε προγράμματα

2. Παράδειγμα με μετρητή

Παράδειγμα. Να γραφεί πρόγραμμα που θα υπολογίζει το μέσο όρο N αριθμών.

```
I=1 ; S=0
```

```
10 READ*,X
```

```
S=S+X
```

```
I=I+1
```

```
IF (I<=N) GOTO 10
```

```
PRINT*, 'MO=',S/N
```



Ανακύκλωση και χρήση μετρητών σε προγράμματα

3. Βήματα επίλυσης προβλημάτων με μετρητές

Βήμα 1. Ζητώ πλήθος επαναλήψεων (N).

Βήμα 2. Αρχικοποίηση παραμέτρων

I=1 (Μετρητής επαναλήψεων)

S=0 (Αθροιστής)

P=1 (Πολλαπλασιαστής)

PL=0 (Μετρητής πλήθους)

Βήμα 3. Διάβασε δεδομένο (X).

Βήμα 4. Επεξεργασία δεδομένων

Χωρίς συνθήκη

S=S+X (Άθροισμα)

P=P*X (Γινόμενο)



Ανακύκλωση και χρήση μετρητών σε προγράμματα

3. Βήματα επίλυσης προβλημάτων με μετρητές (...συνέχεια)

Βήμα 4. Επεξεργασία δεδομένων

Με συνθήκη

IF (συνθήκη) $S=S+X$ (Άθροισμα)

IF (συνθήκη) $P=P*X$ (Γινόμενο)

IF (συνθήκη) $PL=PL+1$ (Πλήθος)

Βήμα 5. Αύξηση του μετρητή πλήθους κατά 1 ($I=I+1$)

Βήμα 6. Έλεγχος αν τελείωσε η διαδικασία

IF ($I \leq N$) GOTO Βήμα 3

Βήμα 7. Εμφάνιση αποτελέσματος.

ΠΡΟΣΟΧΗ. Μέσος όρος = Άθροισμα / Πλήθος



Παράδειγμα στο οποίο δεν απαιτείται κάποια συνθήκη

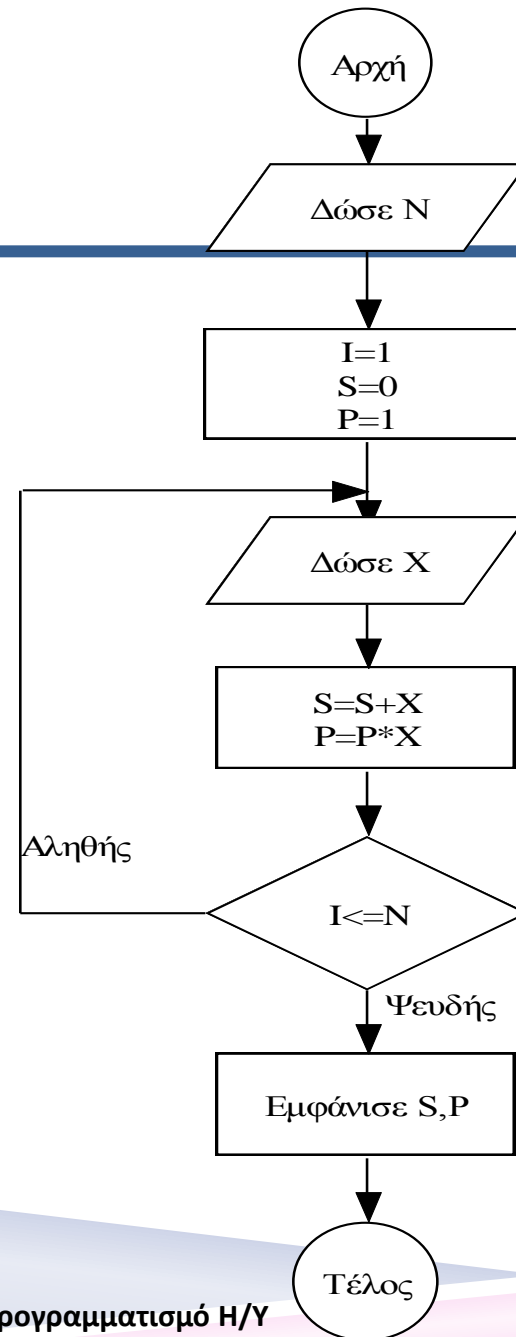
Δίνονται N αριθμοί. Να υπολογιστεί :

A) το άθροισμά τους,

B) το γινόμενό τους.



Αλγόριθμος (χωρίς συνθήκη)



Πρόγραμμα (χωρίς συνθήκη)

```
PROGRAM TEST  
IMPLICIT NONE  
INTEGER          :: I,N  
REAL             :: S,P,X  
PRINT*,'ΔΩΣΕ ΠΛΗΘΟΣ ΑΡΙΘΜΩΝ ='          ! ΒΗΜΑ 1  
READ*,N  
I=1 ; S=0 ; P=1                          ! ΒΗΜΑ 2  
10 PRINT*,'ΔΩΣΕ ΑΡΙΘΜΟ ='                ! ΒΗΜΑ 3  
READ*,X  
S=S+X                                     ! ΒΗΜΑ 4  
P=P*X  
I=I+1                                    ! ΒΗΜΑ 5  
IF (I<=N) GOTO 10                       ! ΒΗΜΑ 6  
PRINT*,'ΑΘΡΟΙΣΜΑ ΑΡΙΘΜΩΝ =',S          ! ΒΗΜΑ 7  
PRINT*,'ΓΙΝΟΜΕΝΟ ΑΡΙΘΜΩΝ=',P  
END PROGRAM TEST
```



Παράδειγμα στο οποίο απαιτείται κάποια συνθήκη

Δίνονται N ακέραιοι αριθμοί. Να υπολογιστεί

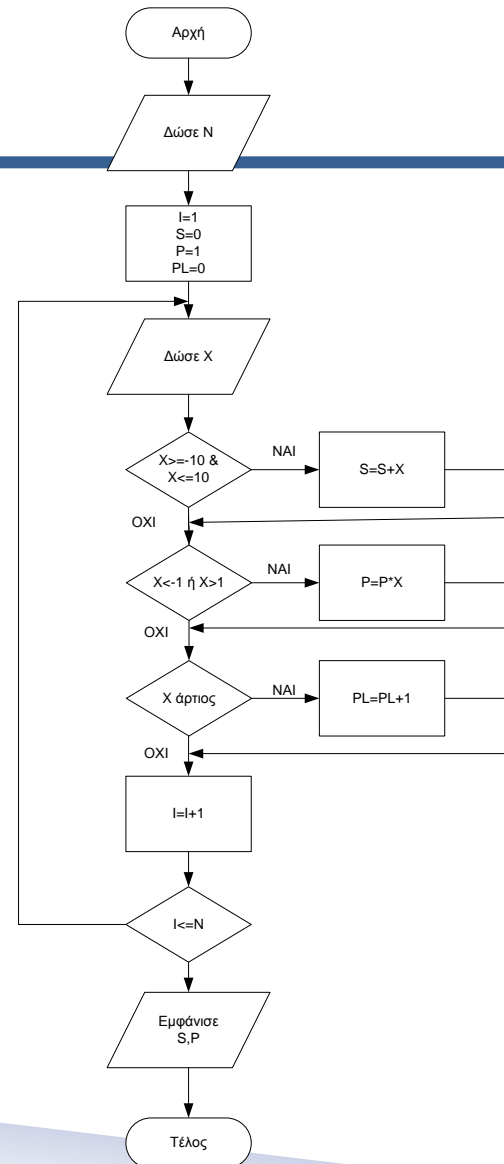
α) το άθροισμα των αριθμών που είναι μέσα στο κλειστό διάστημα $[-10,10]$,

β) το γινόμενο των αριθμών που είναι εκτός του διαστήματος $[-1,1]$,

γ) το πλήθος των αριθμών που είναι άρτιοι.



Αλγόριθμος (με συνθήκη)



Πρόγραμμα (με συνθήκη)

```
PROGRAM TEST  
IMPLICIT NONE  
INTEGER          :: I,N  
REAL             :: S,P,X  
PRINT*,'ΔΩΣΕ ΠΛΗΘΟΣ ΑΡΙΘΜΩΝ ='           ! ΒΗΜΑ 1  
READ*,N  
I=1 ; S=0 ; P=1 ; PL=0                     ! ΒΗΜΑ 2  
10 PRINT*,'ΔΩΣΕ ΑΡΙΘΜΟ ='                 ! ΒΗΜΑ 3  
READ*,X  
IF ((X>=-10).AND.(X<=10)) S=S+X          ! ΒΗΜΑ 4  
IF ((X>1).OR.(X<-1)) P=P*X  
IF (MOD(X,2)==0) PL=PL+1
```



Πρόγραμμα (με συνθήκη) (...συνέχεια)

```
I=I+1
```

```
! ΒΗΜΑ 5
```

```
IF (I<=N) GOTO 10
```

```
! ΒΗΜΑ 6
```

```
PRINT*, 'ΑΘΡΟΙΣΜΑ ΑΡΙΘΜΩΝ ΣΤΟ [-10,10] =', S
```

```
! ΒΗΜΑ 7
```

```
PRINT*, 'ΓΙΝΟΜΕΝΟ ΑΡΙΘΜΩΝ ΕΚΤΟΣ ΤΟΥ [-1,1] =', P
```

```
PRINT*, 'ΠΛΗΘΟΣ ΑΡΤΙΩΝ ΑΡΙΘΜΩΝ =', PL
```

```
END PROGRAM TEST
```



Η εντολή DO – Σύνταξη

Σύνταξη :

[όνομα :] DO [label] “δήλωση”
Σύνολο εντολών
“Δήλωση του τέλους” της DO

“δήλωση” :

Κενό

ή

Μεταβλητή =αρχική τιμή, τελική τιμή [, βήμα]

ή

WHILE (λογική έκφραση)

“δήλωση τέλους” :

END DO [όνομα]

ή

Label εντολή (εκτός από GOTO, RETURN, STOP, EXIT, CYCLE, IF, END).



Η πρώτη μορφή της DO-END DO

Σύνταξη :

DO

 Σύνολο εντολών

END DO

Σκοπός : Η ατέρμονη εκτέλεση ενός συνόλου εντολών.

Παράδειγμα.

 K=0

 DO

 K=K+1

 PRINT*,K

 IF (K>5) EXIT

 END DO



Παραδείγματα της DO-END DO

```
CHARACTER (10) :: USERNAME,PASSWORD
```

```
DO
```

```
    PRINT*, 'USERNAME='
```

```
    READ*, USERNAME
```

```
    PRINT*, 'PASSWORD='
```

```
    READ*, PASSWORD
```

```
    IF ((USERNAME=='NIKOS').AND.(PASSWORD=='12345')) EXIT
```

```
END DO
```

```
DO
```

```
    ! ΑΤΕΡΜΟΝΗ ΑΝΑΚΥΚΛΩΣΗ
```

```
    IF (I>5) EXIT
```

```
    PRINT*, I
```

```
END DO
```

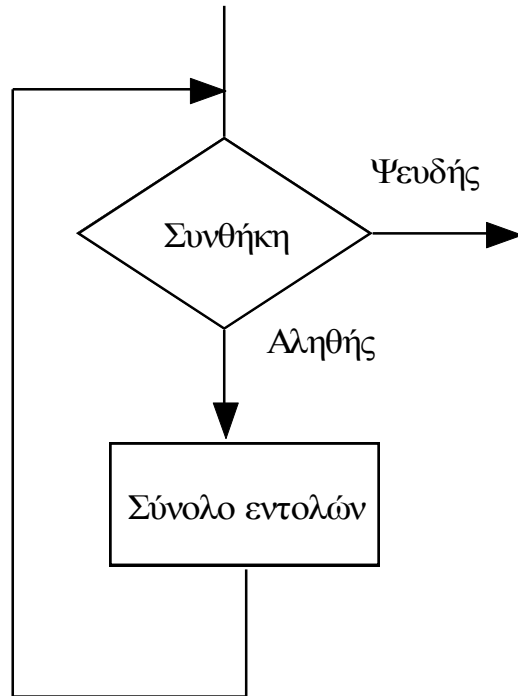


Άσκηση (1)

Να γραφεί πρόγραμμα το οποίο θα διαβάσει ένα σύνολο ακεραίων αριθμών μεταξύ ένα και δέκα και θα υπολογίζει τον μέσο όρο αυτών έως ότου εμφανιστεί ένας αριθμός έξω από τα όρια αυτά.



Η δεύτερη μορφή της DO-END DO



Σύνταξη :

DO WHILE (συνθήκη)

 Σύνολο εντολών

END DO

Σκοπός : Η επανάληψη ενός συνόλου εντολών έως ότου μια συνθήκη ικανοποιηθεί.



Παρατηρήσεις (1)

1. Οι μεταβλητές που εμπλέκονται μέσα στη συνθήκη της WHILE θα πρέπει :
 - α) να πάρουν τιμή πριν τη χρήση της WHILE και
 - β) να αλλάξουν τιμή κατά τη διάρκεια της ανακύκλωσης (μεταξύ DO WHILE και END DO).
2. Η εντολή DO-WHILE χρησιμοποιείται όταν δε γνωρίζουμε εκ των προτέρων το πλήθος των επαναλήψεων.



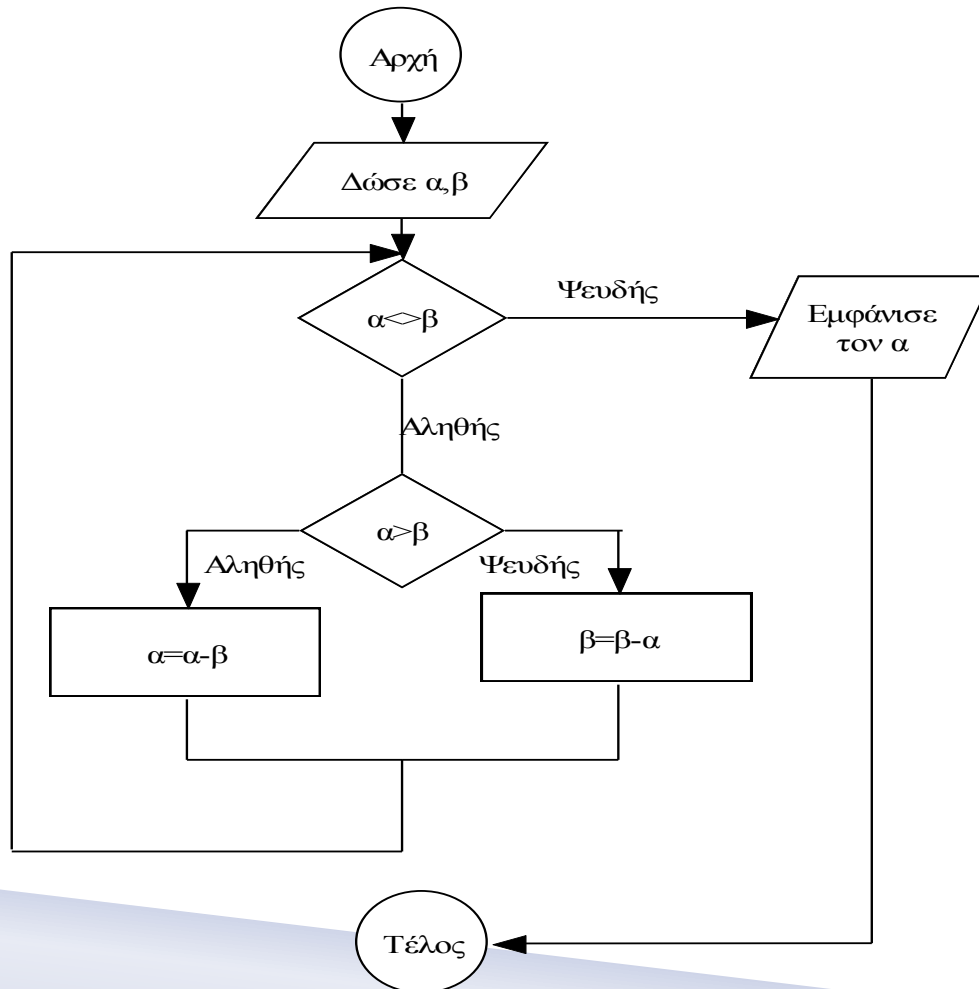
Παράδειγμα 1

Να γραφεί πρόγραμμα το οποίο θα υπολογίζει το μέγιστο κοινό διαιρέτη δύο αριθμών α , β βάσει του αναδρομικού τύπου

$$\text{ΜΚΔ}(\alpha, \beta) = \begin{cases} \text{ΜΚΔ}(\alpha - \beta, \beta) & \text{αν } \alpha > \beta \\ \text{ΜΚΔ}(\alpha, \beta - \alpha) & \text{αν } \alpha < \beta \\ \alpha & \text{αν } \alpha = \beta \end{cases}$$



Αλγόριθμος 1



Πρόγραμμα 1

```
PROGRAM MKD  
IMPLICIT NONE  
INTEGER :: A,B,MKD  
PRINT*,'DOSE DYO AKERAIOYS ARITHMOYS='  
READ*,A,B  
DO WHILE (A/=B)  
        IF (A>B) THEN  
                A=A-B  
        ELSE  
                B=B-A  
        END IF  
END DO  
PRINT*,'MKD=',A  
END PROGRAM MKD
```



Άσκηση «Μέγιστος Κοινός Διαιρέτης (Μ.Κ.Δ.)»

Να γραφεί πρόγραμμα το οποίο θα υπολογίζει το μέγιστο κοινό διαιρέτη δύο αριθμών a , b σύμφωνα με το παρακάτω παράδειγμα π.χ. $ΜΚΔ(12,15)$.

Βήμα 1. Τοποθετώ στην πρώτη θέση τον μεγαλύτερο $ΜΚΔ(15,12)$.

Βήμα 2.

$$\begin{aligned} 15 &= 1 * 12 + 3 && (A=Q_1 B + Y_1) - (Y_1 \neq 0 \rightarrow A < -B, B < -Y_1) \\ 12 &= 4 * 3 + 0 && (B=Q_2 Y_1 + Y_2) - (Y_2 = 0 \rightarrow Y_1 = ΜΚΔ(A,B)) \end{aligned}$$



Παράδειγμα 2

Ο υπολογισμός της τετραγωνικής ρίζας του a γίνεται από τον παρακάτω αναδρομικό τύπο :

$$x = \frac{1}{2} \left(x + \frac{a}{x} \right)$$

Να υπολογιστεί η τετραγωνική ρίζα του a με ακρίβεια ϵ αν πάρουμε ως αρχική τιμή της προσέγγισης $x = \frac{a}{3}$.



Παράδειγμα 2 – Επίλυση με το χέρι

$$a = 1$$

$$x = \frac{a}{3} = \frac{1}{3}$$

$$x_{new} = \frac{1}{2} \left(x + \frac{a}{2} \right) = \frac{1}{2} \left(\frac{1}{3} + \frac{1}{2} \right) = \frac{10}{6}$$

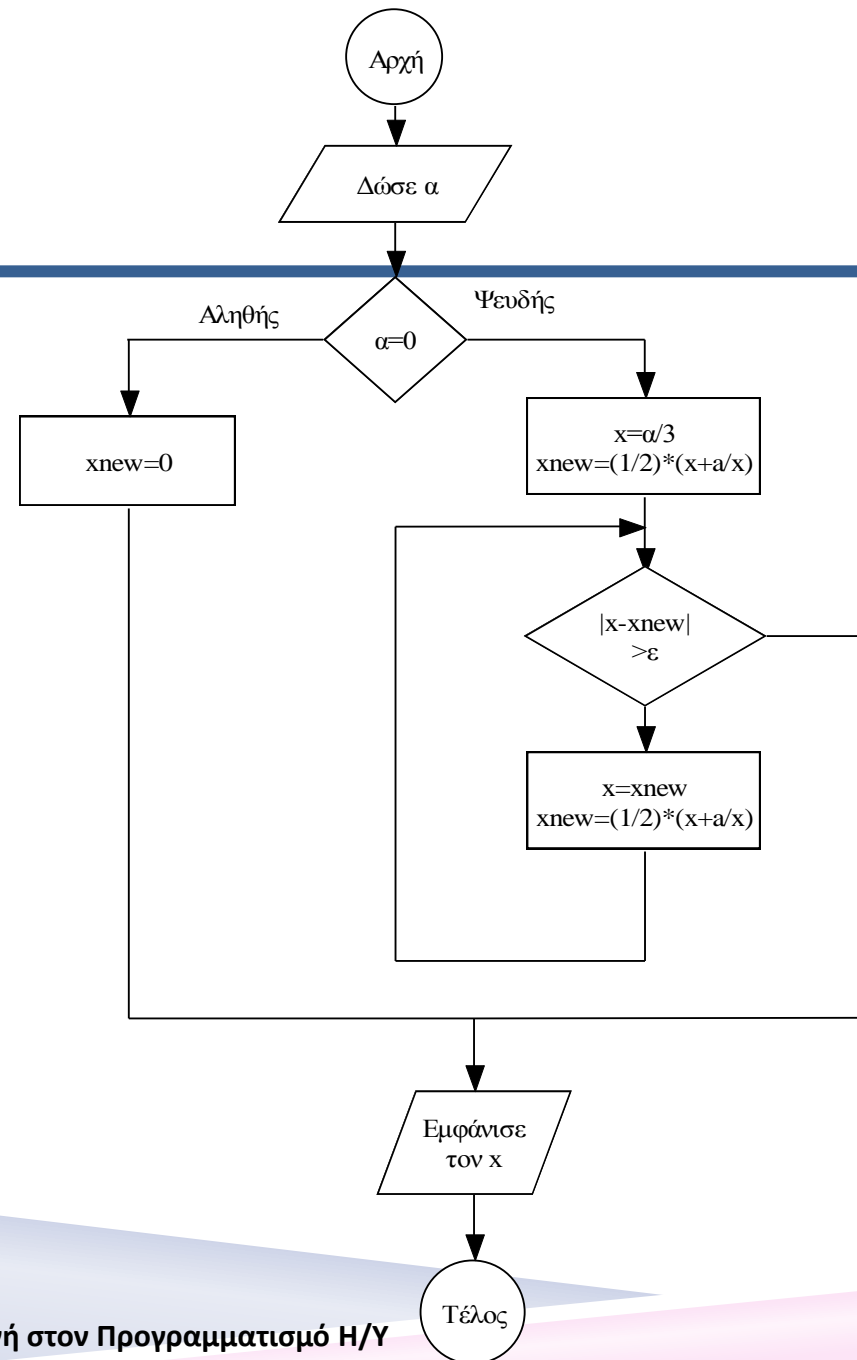
$$|x - x_{new}| = \left| \frac{1}{3} - \frac{10}{6} \right| = \left| \frac{8}{6} \right| > 10^{-6}$$

$$x = x_{new} = \frac{10}{6}, x_{new} = \frac{1}{2} \left(x + \frac{a}{2} \right) = \frac{17}{15}$$

$$|x - x_{new}| = \frac{16}{30} > 10^{-6}$$



Αλγόριθμος 2



Πρόγραμμα 2

```
PROGRAM SQUAREROOT  
IMPLICIT NONE  
REAL                :: A,X,XNEW  
REAL, PARAMETER    :: E=1E-06  
PRINT*,'A=' ; READ*,A  
IF (A==0) THEN  
                XNEW=0  
ELSE  
                X=A/3  
                XNEW=(1/2.0)*(X+A/X)  
                DO WHILE ABS(X-XNEW)>E  
                    X=XNEW  
                    XNEW=(1/2.0)*(X+A/X)  
                END DO  
END IF  
PRINT*,'SQUARE ROOT OF 'A,'IS 'XNEW  
END PROGRAM SQUAREROOT
```



Παράδειγμα 3

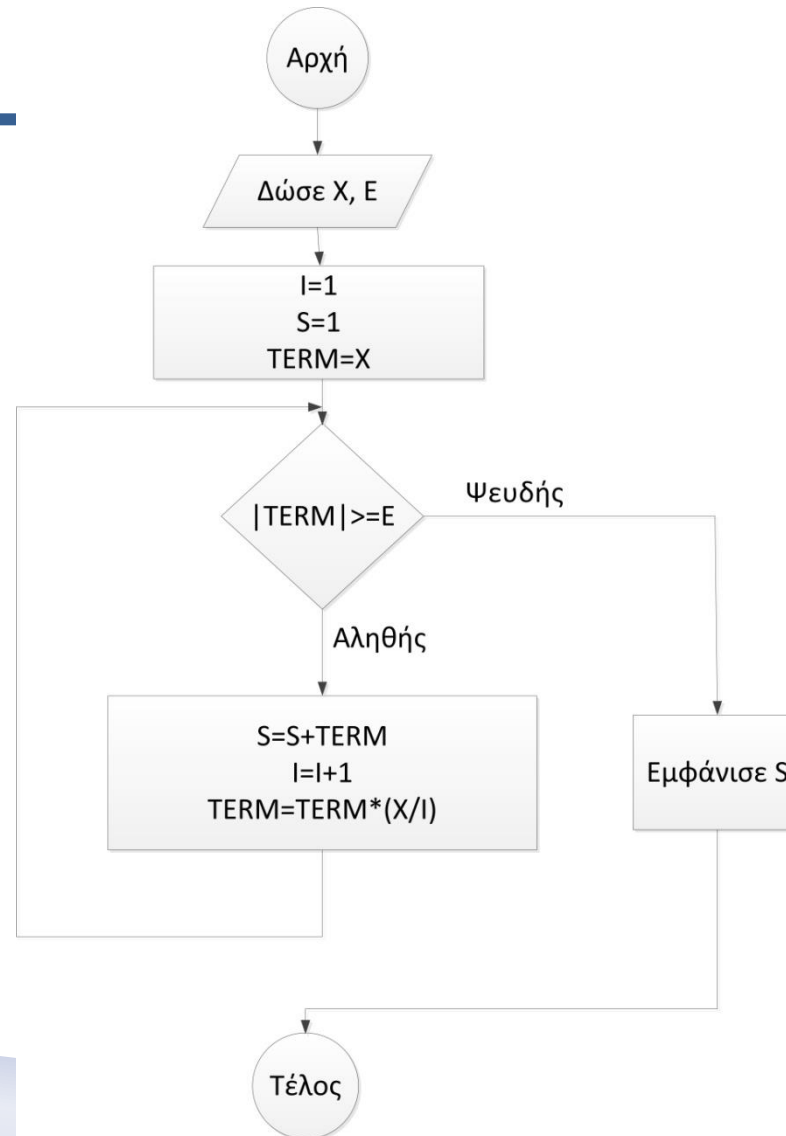
Η εκθετική συνάρτηση $EXP(X)$ ορίζεται ως το άθροισμα των άπειρων όρων της παρακάτω σειράς :

$$EXP(X) = 1 + X + \frac{X^2}{2!} + \frac{X^3}{3!} + \dots + \frac{X^I}{I!} + \dots$$

Να χρησιμοποιήσετε την παραπάνω σειρά για τον υπολογισμό του $EXP(X)$ έως ότου η απόλυτη τιμή του όρου $X^I/I!$ γίνει μικρότερη ή ίση από μια επιθυμητή ακρίβεια E .



Αλγόριθμος 3



Πρόγραμμα 3

```
PROGRAM EXPONENTIAL  
IMPLICIT NONE  
REAL          :: S,TERM,X,E  
INTEGER       :: I  
PRINT*,'X=' ; READ*,X           ! ΕΙΣΑΓΩΓΗ ΔΕΔΟΜΕΝΩΝ  
PRINT*,'E=' ; READ*,E  
I=1                               ! ΑΡΧΙΚΟΠΟΙΗΣΗ ΟΡΩΝ  
S=1  
TERM=X  
DO WHILE (ABS(TERM)>=E)          ! ΠΡΟΣΘΕΣΗ ΟΡΩΝ  
    S=S+TERM  
    I=I+1  
    TERM=TERM*(X/I)  
END DO  
! ΕΜΦΑΝΙΣΗ ΤΩΝ ΑΠΟΤΕΛΕΣΜΑΤΩΝ  
PRINT*,'EXP('X,')='S  
END PROGRAM EXPONENTIAL
```



Άσκηση (2)

Να υπολογίσετε που συγκλίνει η σειρά :

$$S = \frac{1}{1!} + \frac{1}{2!} + \frac{1}{3!} + \dots + \frac{1}{n^2} + \dots$$

Σύμφωνα με τον Euler (1730) η παραπάνω σειρά συγκλίνει στο $\pi^2/6$ και κατά συνέπεια μπορεί να χρησιμοποιηθεί στον υπολογισμό του π .



Η τρίτη μορφή της DO-END DO

Σύνταξη :

DO μεταβλητή(M)=αρχική τιμή(A), τελική τιμή(T) [,βήμα(B)]

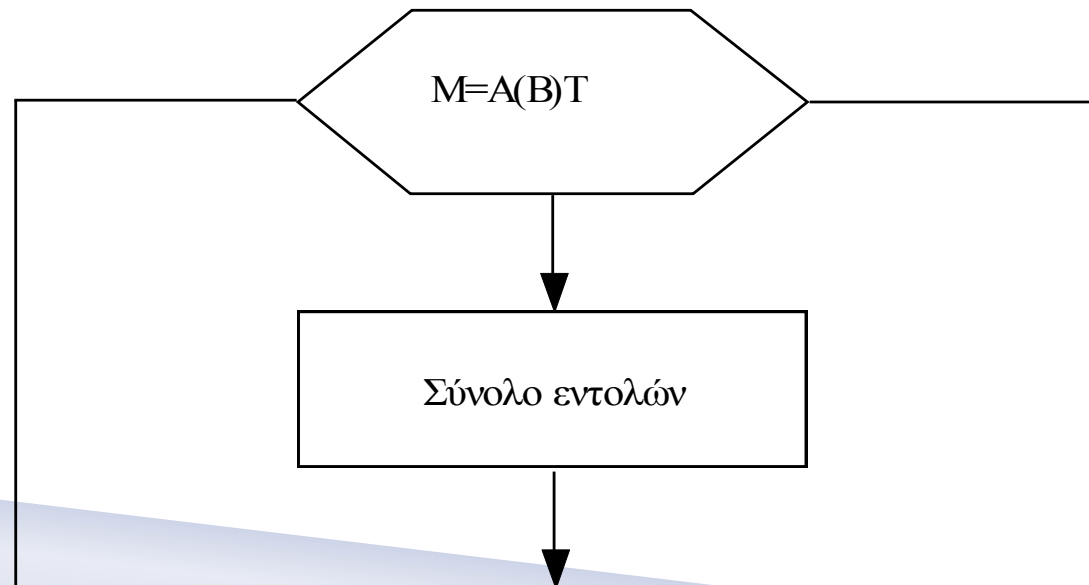
 Σύνολο εντολών

END DO

DO label μεταβλητή(M)=αρχική τιμή(A), τελική τιμή(T) [,βήμα(B)]

 Σύνολο εντολών

Label CONTINUE



Παραδείγματα της τρίτης μορφή της DO-END DO

```
DO I=1,5,1
    PRINT*,I           1
END DO                2
    DO 10 I=1,5,1     3
        PRINT*,I     4
10    CONTINUE       5
```

Άσκηση. Να γραφεί πρόγραμμα που θα εκτυπώνει τους αριθμούς από το 1 ως το 5 ανάποδα, δηλ. 5,4,3,2,1.



Παρατηρήσεις (2)

1. Το βήμα θα πρέπει να είναι πάντα διαφορετικό από το μηδέν.
2. Δεν αλλάζουν οι τιμές του βήματος, αρχικής και τελικής τιμής μέσα στο loop.
3. Να μην παραλείψουμε το βήμα όταν είναι αρνητικό.
4. Η μεταβλητή θα πρέπει να είναι ακέραια όπως και οι αρχική τιμή, τελική τιμή και βήμα.
5. Είναι λάθος να χρησιμοποιείς μια GO TO για να σε μεταφέρει μέσα σε μια DO.
6. Αριθμός επαναλήψεων σε μια DO-END DO.

$$\left[\frac{(\text{αρχική}) - (\text{τελική})}{\text{βήμα}} \right] + 1$$



Παρατηρήσεις (2) (...συνέχεια)

7. Σταμάτημα της εκτέλεσης ΜΙΑΣ επανάληψης (CYCLE)
IF (X>5) CYCLE
8. Έξοδος από μια DO-END DO (EXIT)
IF (X>5) EXIT
9. Προσοχή πρέπει να δίνεται στις μεταβολές της «μεταβλητής».

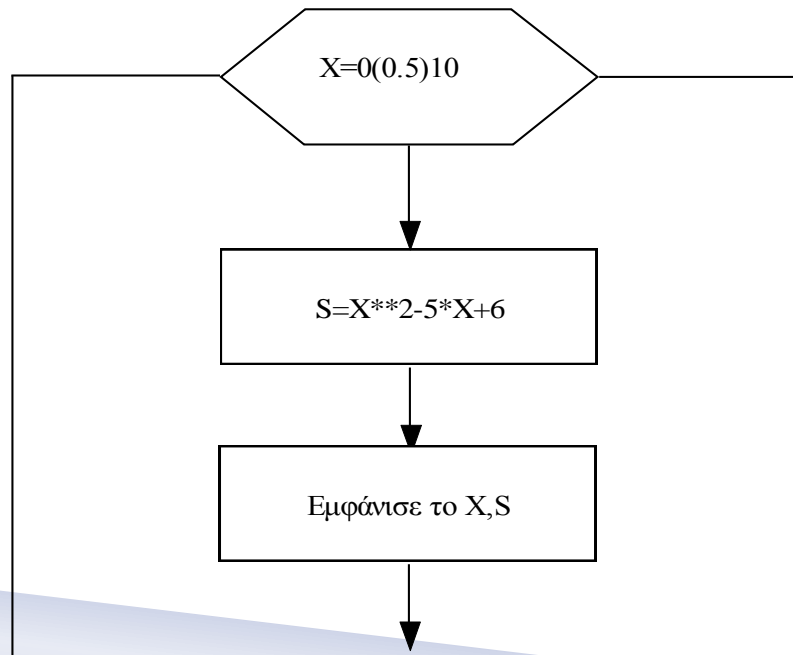


Παράδειγμα (Πολυώνυμο)

Να υπολογιστούν οι τιμές του πολυωνύμου

$$P(X) = X^2 - 5X + 6$$

για $X=0,0.5,1,\dots,10$.



Πρόγραμμα 1 (Πολυώνυμο)

```
PROGRAM POLYONYMO  
IMPLICIT NONE  
REAL   :: X,S  
DO X=0,10,0.5  
  
        S=X**2-5*X+6  
        WRITE(6,10) S,X  
10      FORMAT(1X,F5.1,5X,F8.2)  
END DO  
END PROGRAM POLYONYMO
```



Πρόγραμμα 2 (Πολυώνυμο)

```
PROGRAM POLYONOMO  
IMPLICIT NONE  
INTEGER:: I  
REAL    :: X,S  
DO I=0,100,5  
    X=0.1*I  
    S=X**2-5*X+6  
    WRITE(6,10) S,X  
10  FORMAT(1X,F5.1,5X,F8.2)  
END DO  
END PROGRAM POLYONOMO
```

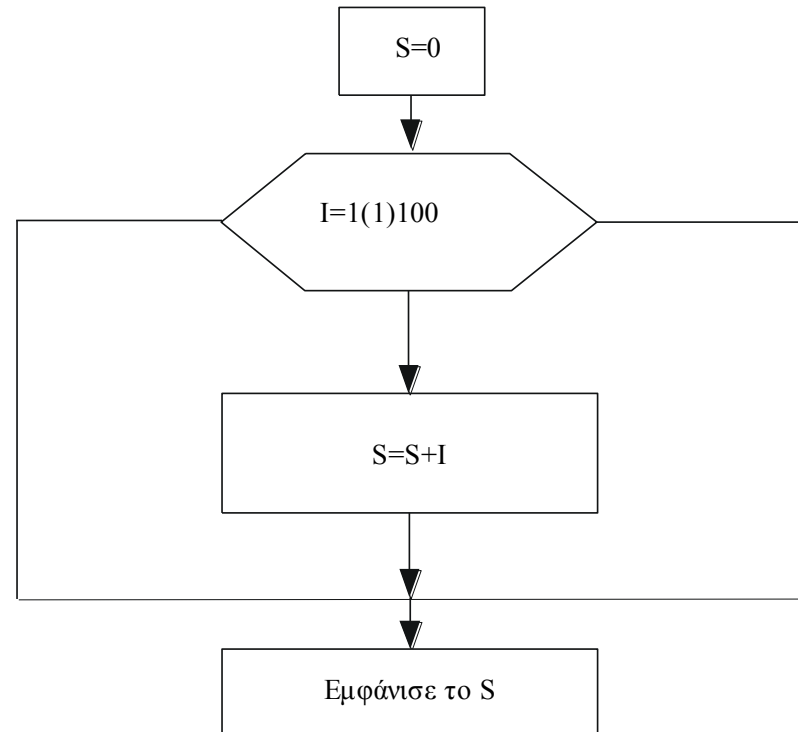
Να υπολογιστούν οι τιμές του πολυωνύμου για $X=0, 0.05, 0,10, \dots,10$.



Παράδειγμα (Άθροισμα)

Να υπολογισθεί το άθροισμα

$$S = 1 + 2 + 3 + \dots + 100$$



Πρόγραμμα (Άθροισμα)

```
PROGRAM SUM  
IMPLICIT NONE  
INTEGER          :: S,I  
DO I=1,100  
    S=S+I  
END DO  
PRINT*,'S=',S  
END PROGRAM SUM
```

Όμοια για $\sum_{i=a}^b f(i)$, $step = c$.



Άσκηση (3)

Να γραφεί πρόγραμμα που θα υπολογίζει το παρακάτω άθροισμα

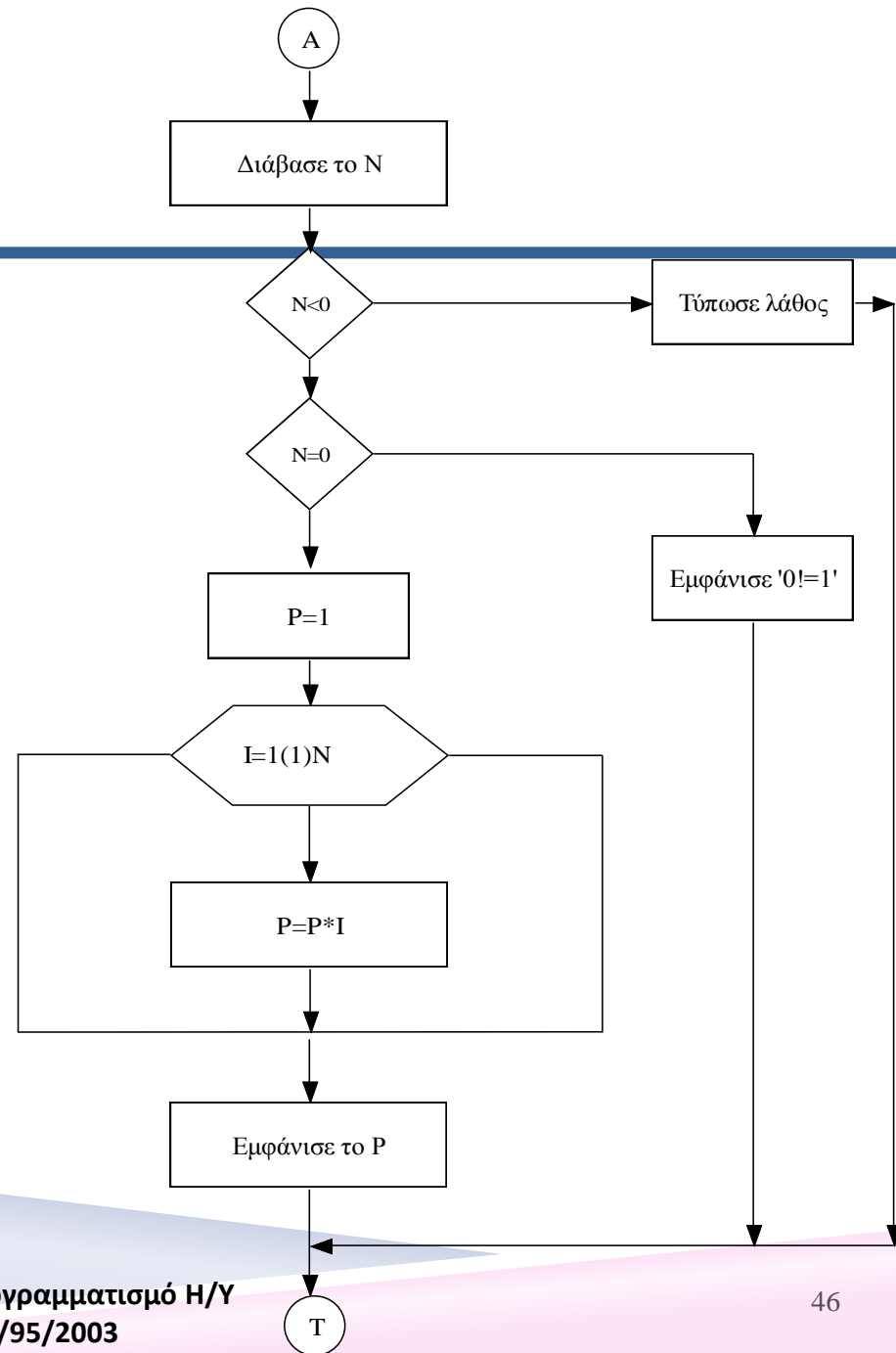
$$S = 1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \dots + \frac{1}{41} - \frac{1}{43} = \sum \left(\frac{1}{N} - \frac{1}{N+2} \right),$$
$$N = 1, 5, 9, \dots, 41$$

Η παραπάνω σειρά συγκλίνει πολύ αργά προς το $\pi=3.14$.



Παράδειγμα 5

Να υπολογιστεί το $N!$
π.χ. $N! = 1 * 2 * 3 * .. * N$.



Πρόγραμμα 5

```
PROGRAM FACTORIAL
```

```
IMPLICIT NONE
```

```
INTEGER*4      :: I,N,P
```

```
PRINT*,'N=' ; READ*,N
```

```
IF (N<0) THEN
```

```
    PRINT*,'ERROR, N MUST BE POSITIVE'
```

```
ELSE IF (N==0) THEN
```

```
    PRINT*,'0!=1'
```

```
ELSE
```



Πρόγραμμα 5 (...συνέχεια)

```
P=1
  DO I=1,N
    P=P*I
  END DO
  PRINT*,N,'!= ',P
END IF
END PROGRAM FACTORIAL
```

Όμοια για

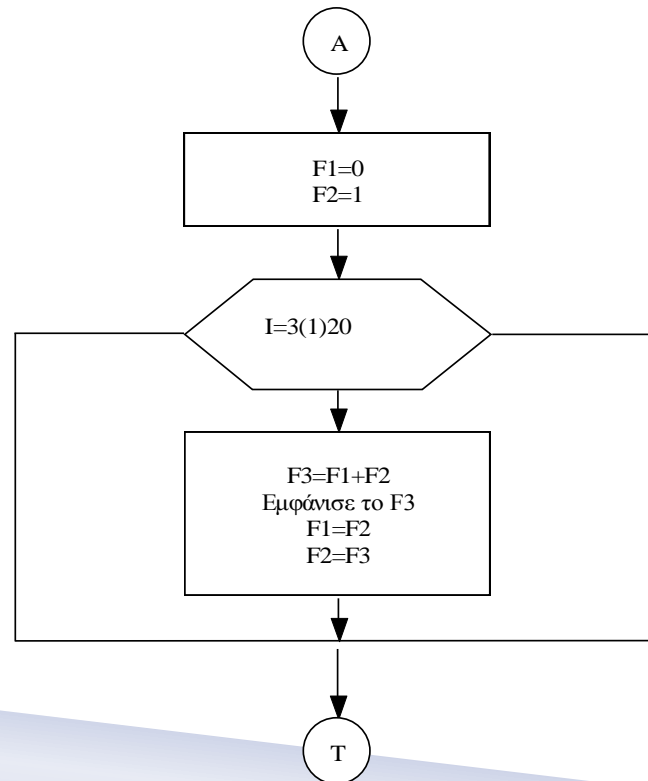
$$\prod_{i=a}^b f(i), step = c, P = \prod_{n=1}^{20} \frac{(n+1)^2}{n(n+2)}$$



Παράδειγμα (Ακολουθία Fibonacci)

Να βρεθούν οι 20 πρώτοι όροι της ακολουθίας Fibonacci :

$$F_N = F_{N-1} + F_{N-2}, N > 2 \text{ με } F_1 = F_2 = 1$$



Πρόγραμμα (Ακολουθία Fibonacci)

```
PROGRAM FIBONACCI  
IMPLICIT NONE  
INTEGER :: F1,F2,F3,N  
F1=1 ; F2=1  
DO N=3,20  
    F3=F1+F2  
    PRINT*,N,F3  
    F1=F2 ; F2=F3  
END DO  
END PROGRAM FIBONACCI
```

Στο παραπάνω πρόγραμμα να εκτυπώσετε τον λόγο : $\frac{F_n}{F_{n-1}}$
ο οποίος συγκλίνει στη χρυσή αναλογία $\phi=1.61803\dots$



Παράδειγμα 6

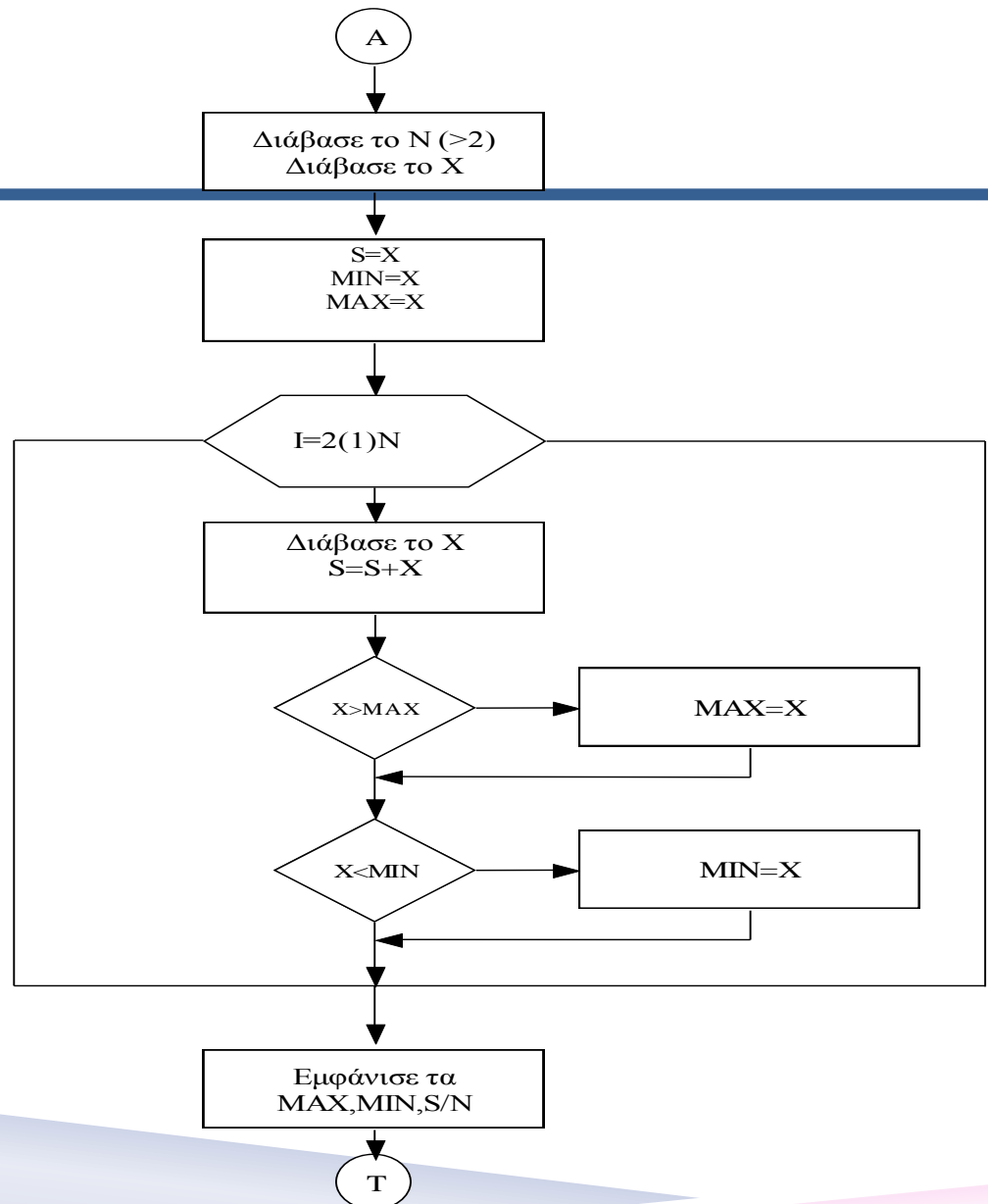
Να διαβασθούν $N > 2$ αριθμοί και, αφού υπολογισθούν, να εμφανιστούν τα εξής :

α) ο μεγαλύτερος αριθμός, β) ο μικρότερος αριθμός, γ) ο μέσος όρος

$$\mu = \frac{\sum_{i=1}^N X_i}{N}$$



Αλγόριθμος 6



Πρόγραμμα 6

```
PROGRAM MINMAXSUM  
IMPLICIT NONE  
REAL          :: S,X,MIN,MAX  
INTEGER       :: I,N  
PRINT*,'N (N>2) =' ; READ*,N  
PRINT*,'X=' ; READ*,X  
S=X  
MIN=X  
MAX=X
```



Πρόγραμμα 6 (...συνέχεια)

```
DO I=2,N
    PRINT*,'X=' ;    READ*,X
    IF (X>MAX) MAX=X
    IF (X<MIN) MIN=X
    S=S+X
END DO
PRINT*,'MINIMUM=',MIN
PRINT*,'MAXIMUM=',MAX
PRINT*,'MEAN=',S/N
END PROGRAM MINMAXSUM
```



Παράδειγμα 1

(Φωλιασμένες εντολές επανάληψης)

```
I=1
DO J=1,10
  PRINT*,I,'*',J,'=',I*J
END DO
I=2
DO J=1,10
  PRINT*,I,'*',J,'=',I*J
END DO
```

```
1*1= 1
1*2= 2
1*3= 3
1*4= 4
1*5= 5
1*6= 6
1*7= 7
1*8= 8
1*9= 9
1*10= 10
2*1= 2
2*2= 4
2*3= 6
2*4= 8
2*5= 10
2*6= 12
2*7= 14
2*8= 16
2*9= 18
2*10= 20
```

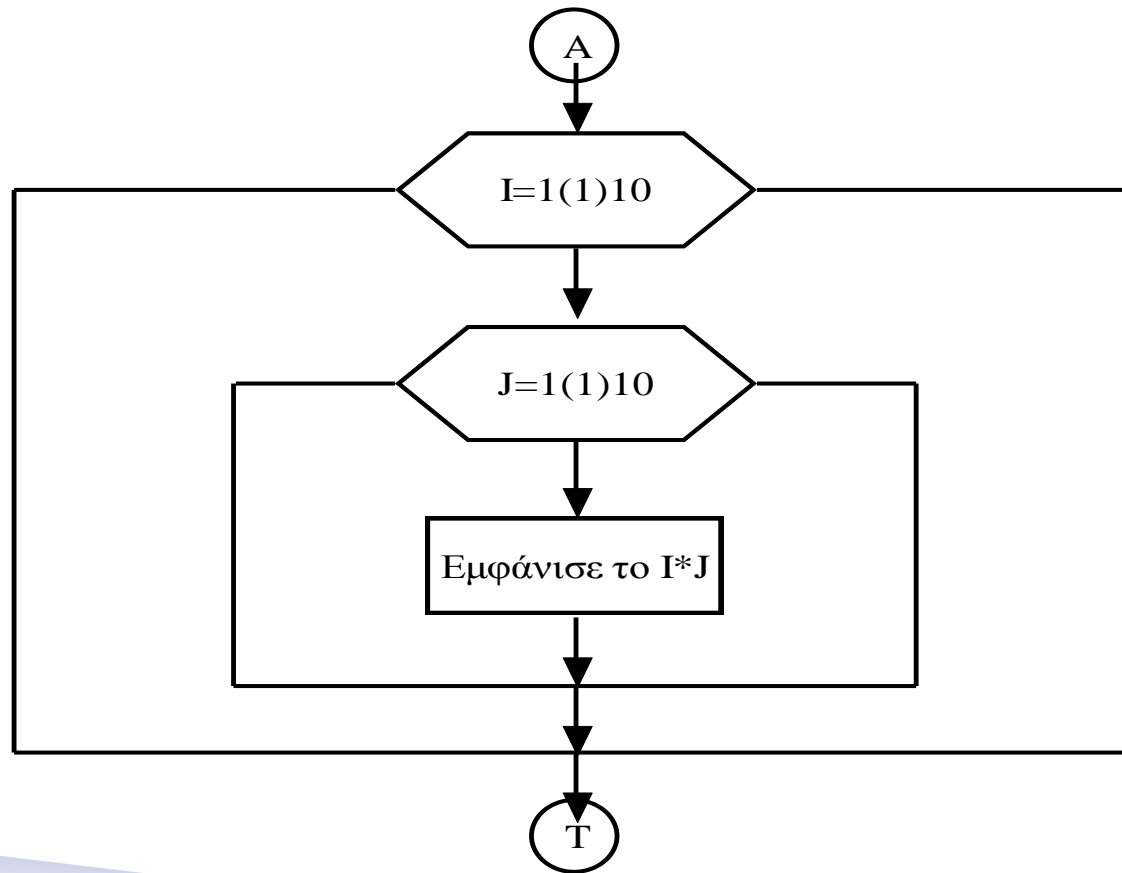
Press any key to continue

Press any key to continue



Αλγόριθμος 1

(Φωλιασμένες εντολές επανάληψης)



Πρόγραμμα 1

(Φωλιασμένες εντολές επανάληψης)

```
PROGRAM PROPAAIDEIA  
IMPLICIT NONE  
INTEGER      ::I,J  
DO I=1,10  
    DO J=1,10  
        PRINT*,I,'*',J,'=',I*J  
    END DO  
    PRINT*,'-----'  
END DO  
END PROGRAM PROPAAIDEIA
```



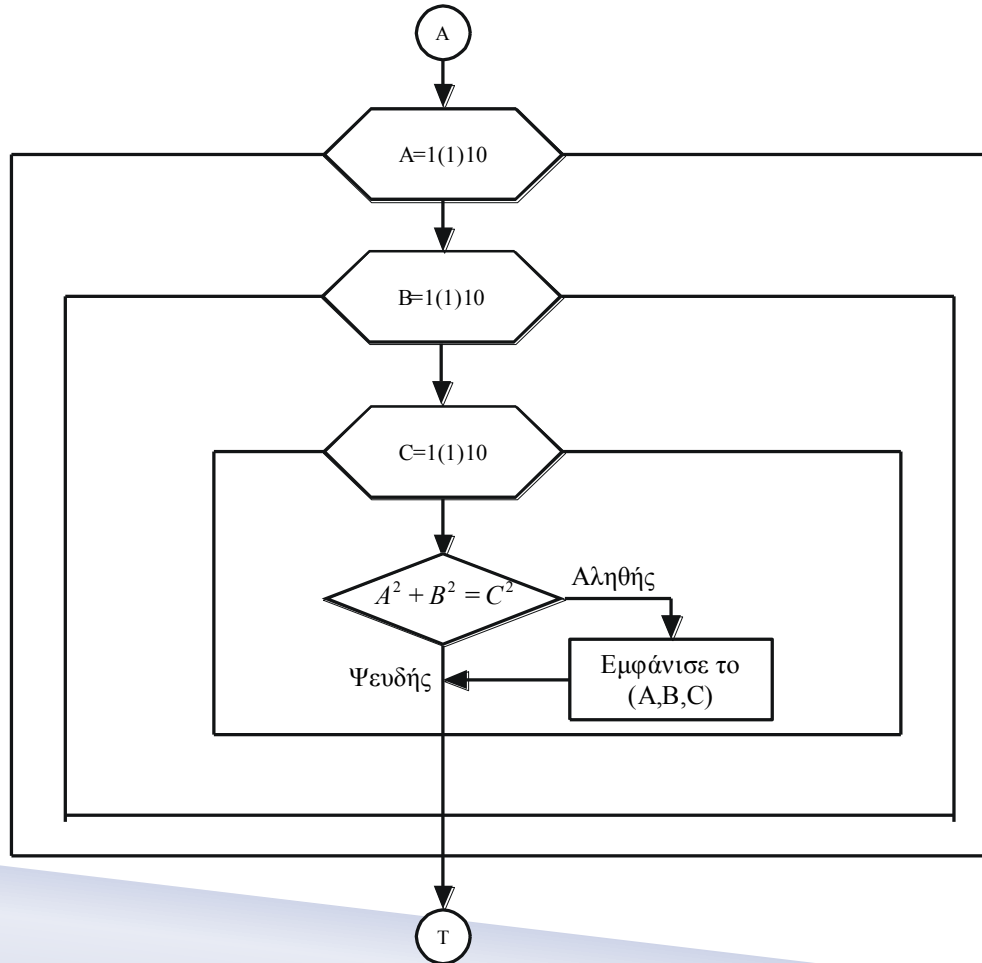
Παράδειγμα (Πυθαγόρειοι Αριθμοί) (Φωλιασμένες εντολές επανάληψης)

Να βρεθούν οι ακέραιοι αριθμοί A,B,C (Πυθαγόρειοι αριθμοί) που ανήκουν στο διάστημα [1,10] και ικανοποιούν την ιδιότητα :

$$A^2 + B^2 = C^2, 3^2 + 4^2 = 5^2$$



Αλγόριθμος (Πυθαγόρειοι Αριθμοί)



Πρόγραμμα 1 (Πυθαγόρειοι Αριθμοί)

```
PROGRAM PYTHAGORIOI
```

```
IMPLICIT NONE
```

```
INTEGER :: A,B,C
```

```
META: DO A=1,10
```

```
METB: DO B=1,10
```

```
METC: DO C=1,10
```

```
IF (A**2+B**2==C**2) THEN
```

```
PRINT*,A,B,C
```

```
END IF
```

```
END DO METC
```

```
END DO METB
```

```
END DO META
```

```
END PROGRAM PYTHAGORIOI
```

3 4 5

4 3 5

6 8 10

8 6 10

Press any key to continue



Πρόγραμμα 2 (Πυθαγόρειοι Αριθμοί)

```
PROGRAM PYTHAGORIOI
IMPLICIT NONE
INTEGER      :: A,B,C
META: DO A=1,10
METB:      DO B=A+1,10
METC:      DO C=B+1,10
                IF (A**2+B**2==C**2) THEN
                    PRINT*,A,B,C
                END IF
            END DO METC
        END DO METB
    END DO META
END PROGRAM PYTHAGORIOI
```

3 4 5
6 8 10

Press any key to continue



Πρόγραμμα 3 (Πυθαγόρειοι Αριθμοί)

```
PROGRAM PYTHAGORIOI
IMPLICIT NONE
INTEGER      :: A,B,C
META: DO A=1,8
METB:      DO B=A+1,9
METC:      DO C=B+1,10
              IF (A**2+B**2==C**2) THEN
                  PRINT*,A,B,C
              END IF
            END DO METC
          END DO METB
        END DO META
END PROGRAM PYTHAGORIOI
```

3 4 5
6 8 10

Press any key to continue



Σύνολο επαναλήψεων

A	B	C	Επαναλήψεις
1	2	3,4,...,N	N-2
1	3	4,5,...,N	N-3
...
1	N-1	N	1
Σύνολο Επαναλήψεων			1+2+...+(N-2)
2	3	4,5,...,N	N-3
2	4	5,6,...,N	N-2
...
2	N-1	N	1
Σύνολο Επαναλήψεων			1+2+...+(N-3)
...
N-2	N-1	N	1

$$\begin{aligned}
 & (1 + 2 + \dots + (N - 2)) + (1 + 2 + \dots + (N - 3)) + \dots + 1 = \\
 & = 1 \times (N - 2) + 2 \times (N - 3) + 3 \times (N - 4) + \dots + (N - 2) \times 1 = \\
 & = \sum_{i=1}^{N-2} i \times (N - i - 1) = \frac{1}{6} N(N - 1)(N - 2)
 \end{aligned}$$



Ασκήσεις

Άσκηση 1

Να βρεθούν οι ακέραιοι αριθμοί X, Y που ανήκουν στο διάστημα $[1, 10]$ και ικανοποιούν την ιδιότητα :

$$X^2 - 5Y^2 + 1 = 0$$

Άσκηση 2

Η ετήσια δόση X για την εξόφληση δανείου ποσού D , το οποίο πρέπει να εξοφληθεί σε N χρόνια με επιτόκιο E ($0 < E < 1$) δίνεται από τον τύπο :

$$X = \frac{D(1 + E)^N E}{(1 + E)^N - 1}$$

Αν το ποσό είναι $D = 1.000.000$, να υπολογιστεί η ετήσια δόση για επιτόκιο $E = 0.05, 0.055, 0.06, 0.065, 0.07, 0.075$ και σε χρόνια $N = 10, 15, 20, 25, 30$.



Βιβλιογραφία

Ν. Καραμπετάκης, *Εισαγωγή στη Fortran 90/95/2003*, 2^η Έκδοση, Θεσσαλονίκη: Εκδόσεις Ζήτη, 2011.



Σημείωμα Αναφοράς

Copyright Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης, Καραμπετάκης Νικόλαος. «Εισαγωγή στον Προγραμματισμό Η/Υ (Fortran 90/95/2003). Εντολές Επανάληψης». Έκδοση: 1.0. Θεσσαλονίκη 2014. Διαθέσιμο από τη δικτυακή διεύθυνση: <http://eclass.auth.gr/courses/OCRS145/>.



Σημείωμα Αδειοδότησης

Το παρόν υλικό διατίθεται με τους όρους της άδειας χρήσης Creative Commons Αναφορά, Παρόμοια Διανομή 4.0 [1] ή μεταγενέστερη, Διεθνής Έκδοση. Εξαιρούνται τα αυτοτελή έργα τρίτων π.χ. φωτογραφίες, διαγράμματα κ.λ.π., τα οποία εμπεριέχονται σε αυτό και τα οποία αναφέρονται μαζί με τους όρους χρήσης τους στο «Σημείωμα Χρήσης Έργων Τρίτων».



[1] <http://creativecommons.org/licenses/by-sa/4.0/>





Τέλος Ενότητας

Επεξεργασία: Αναστασία Γ. Γρηγοριάδου
Θεσσαλονίκη, Χειμερινό Εξάμηνο 2013-2014

