

## ΕΝΟΤΗΤΑ 1: Εισαγωγή στους Η/Υ

### Μάθημα 1.4: Το Διάγραμμα Ροής (Flow Chart)

Δημήτρης Ψούνης



## Περιεχόμενα Μαθήματος

### Α. Θεωρία

#### 1. Διάγραμμα Ροής Προγράμματος

1. Το Διάγραμμα Ροής Προγράμματος (ΔΡΠ)
2. Δομικά στοιχεία του ΔΡΠ
3. Είσοδος - Έξοδος
4. Εντολές
5. Δομή Συνθήκης
  1. Σύνθετη Δομή Συνθήκης
6. Δομή Επανάληψης
  1. Εμφωλιασμένοι Βρόχοι
  2. Αμυντικός Προγραμματισμός
7. Πίνακες

### Ασκήσεις

### Α. Θεωρία

#### 1. Διάγραμμα Ροής Προγράμματος

##### 1. Το Διάγραμμα Ροής Προγράμματος

- Το **Διάγραμμα Ροής Προγράμματος (ΔΡΠ)** είναι ένα σχέδιο για την λύση ενός προβλήματος που αποτελείται από μία σειρά από σαφώς καθορισμένα βήματα
  - Αποτελεί δηλαδή ένα σχέδιο που υλοποιεί έναν **αλγόριθμο** επίλυσης ενός προβλήματος
- Ένας **αλγόριθμος** (κατά τον Donald Knuth) είναι ένας επιλύτης ενός προβλήματος με τα εξής χαρακτηριστικά:
  - **Ακρίβεια:** Τα βήματα πρέπει να είναι σαφή.
  - **Μοναδικότητα:** Τα ενδιάμεσα αποτελέσματα είναι μοναδικά για κάθε είσοδο (δεν μπορεί να προκύψουν διαφορετικά ενδιάμεσα αποτελέσματα για την ίδια είσοδο)
  - **Αριθμός Βημάτων:** Πρέπει να είναι πεπερασμένα. Δηλαδή πρέπει ο αλγόριθμος κάποια στιγμή να τελειώνει, μετά από πεπερασμένο αριθμό βημάτων.
  - **Γενικότητα:** Ο αλγόριθμος πρέπει να λειτουργεί για όλες τις εισόδους ενός συγκεκριμένου τύπου.
  - **Είσοδος-Έξοδος:** Ο αλγόριθμος πρέπει να παίρνει κάποια είσοδο και να παράγει μία μοναδική έξοδο για κάθε είσοδο.

#### Δηλαδή:

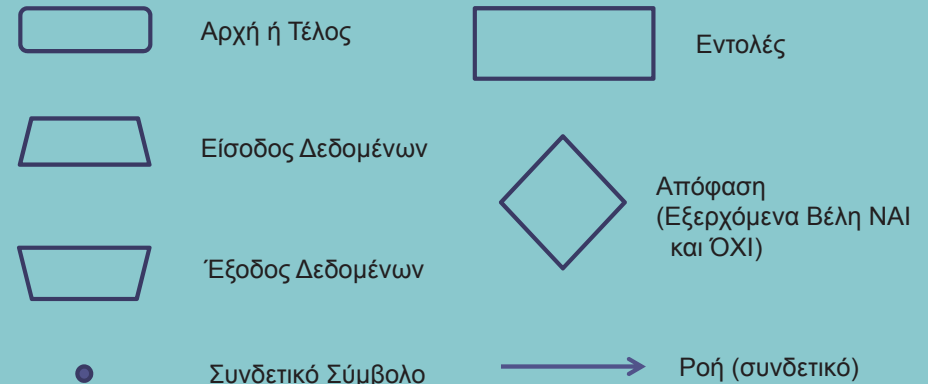
Η δουλειά μας είναι να ορίζουμε αυτοματοποιημένες διαδικασίες που λύνουν το πρόβλημα (κατασκευή αλγορίθμων) και το ΔΡΠ απεικονίζει τη διαδικασία αυτή σχηματικά.

### Α. Θεωρία

#### 1. Διάγραμμα Ροής Προγράμματος

##### 2. Δομικά Στοιχεία του ΔΡΠ

- Ένα **Διάγραμμα Ροής Προγράμματος (ΔΡΠ)** αποτελείται από τα εξής δομικά στοιχεία:



**Παρατήρηση:** Ένα Συμπληρωμένο Διάγραμμα Ροής είναι μια βοήθεια για τον προγραμματισμό. Μεθοδολογικά μία κατασκευή ενός προγράμματος αποτελείται από τρία βήματα: (α) Δημιουργία του Αλγορίθμου, (β) Δημιουργία του Διαγράμματος (είτε ΔΡΠ είτε άλλο) (γ) Κατασκευή του προγράμματος σε γλώσσα προγραμματισμού.

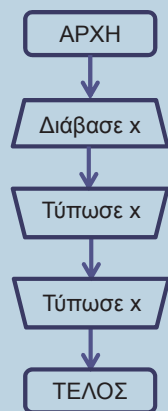
## A. Θεωρία

### 1. Διάγραμμα Ροής Προγράμματος

#### 3. Είσοδος / Έξοδος

Υποθέτουμε ότι το **Διάγραμμα Ροής** είναι το **Πρόγραμμα** που τρέχει στον Υπολογιστή και ότι η **είσοδος** είναι από το πληκτρολόγιο και η **έξοδος** είναι στην οθόνη. Έτσι τα δύο πρώτα δομικά στοιχεία του ΔΡΠ είναι για την είσοδο και την έξοδο του προγράμματος.

**Παράδειγμα:** Το ακόλουθο ΔΡΠ προδιαγράφει έναν αλγόριθμο που διαβάζει από το πληκτρολόγιο έναν αριθμό και έπειτα τον τυπώνει 2 φορές στην οθόνη.



**Παρατήρηση:** Η μεταβλητή είναι αποθηκευτικός χώρος στον υπολογιστή που συγκρατεί την τιμή της μεταβλητής

#### Παραδείγματα (εκτέλεσης):

Είσοδος: 15  
Έξοδος: 15 15

Είσοδος: 8090  
Έξοδος: 8090 8090

## A. Θεωρία

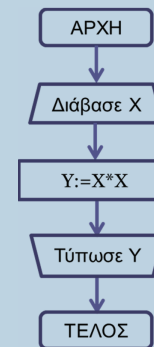
### 1. Διάγραμμα Ροής Προγράμματος

#### 4. Εντολές

Οι **εντολές** είναι προγραμματιστικές διαχειρίσεις μεταβλητών. Η πιο συχνή είναι η **εντολή καταχώρησης** (εκχώρησης) σε διάφορες παραλλαγές:

- $X:=5$  **Απλή Καταχώρηση** (αποθήκευσε την τιμή 5 στην μεταβλητή X)
- $X:=4+2$  **Υπολογισμός και καταχώρηση** (κάνε τον υπολογισμό στα δεξιά του  $:=$  και έπειτα αποθήκευσε το αποτέλεσμα στην X)

**Παράδειγμα:** Το ακόλουθο ΔΡΠ προδιαγράφει έναν αλγόριθμο που διαβάζει από το πληκτρολόγιο έναν αριθμό και έπειτα τυπώνει το τετράγωνό του



#### Παραδείγματα (εκτέλεσης):

Είσοδος: 8  
Έξοδος: 64

Είσοδος: 100  
Έξοδος: 10000

## A. Θεωρία

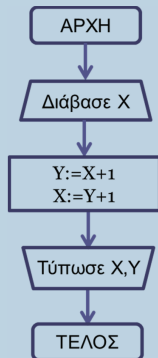
### 1. Διάγραμμα Ροής Προγράμματος

#### 4. Εντολές

Οι **εντολές** μπορούν να γίνουν αρκετά σύνθετες. Είναι σημαντικό ότι μία μεταβλητή είναι ένας αποθηκευτικός χώρος με μία τιμή και ότι γίνεται πρώτα η πράξη στα δεξιά του  $:=$  και έπειτα η εκχώρηση:

- $X:=Y+1$  (αποθήκευσε το αποτέλεσμα της πράξης  $Y+1$  στην μεταβλητή X)
- $X:=X+2$  (αποθήκευσε το αποτέλεσμα της πράξης  $X+2$  στην μεταβλητή X)

**Παράδειγμα:** Το ακόλουθο ΔΡΠ προδιαγράφει έναν αλγόριθμο που διαβάζει από το πληκτρολόγιο έναν αριθμό και έπειτα κάνει έναν (χωρίς συγκεκριμένο λόγο) υπολογισμό.



#### Παραδείγματα (εκτέλεσης):

Είσοδος: 8  
Έξοδος: 10,9

Είσοδος: 100  
Έξοδος: 102,101

## A. Θεωρία

### 1. Διάγραμμα Ροής Προγράμματος

#### 4. Εντολές

Άσκηση: Κατασκευάστε Διάγραμμα Ροής Προγράμματος που διαβάζει δύο αριθμούς και τυπώνει το άθροισμα των τετραγώνων τους.

## A. Θεωρία

### 1. Διάγραμμα Ροής Προγράμματος

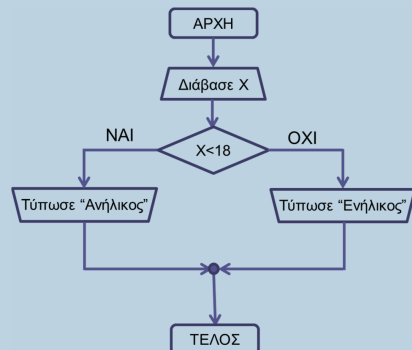
#### 5. Δομή Συνθήκης

Η συνθήκη είναι πολύ σημαντικό προγραμματιστικό εργαλείο. Κάνει τον έλεγχο (π.χ. σύγκριση αριθμητικών τιμών και επιστρέφει Αληθές ή Ψευδές. Εδώ μελετάμε συνήθως σχέσεις ανισοτήτων.

• Π.χ.  $5 > 3$ : Επιστρέφει Αληθές και  $4 < 2$ : Επιστρέφει Ψευδές

Στο ΔΡΠ πρέπει να υπάρχει ροή από το Αληθές και το Ψευδές (βάζουμε στην αρχή του βέλους ΝΑΙ και ΟΧΙ αντίστοιχα)

**Παράδειγμα:** Το ακόλουθο ΔΡΠ προδιαγράφει έναν αλγόριθμο που διαβάζει την ηλικία ενός ατόμου και τυπώνει αν είναι ανήλικος ή ενήλικος.



#### Παραδείγματα (εκτέλεσης):

Είσοδος: 14  
Έξοδος: Ανήλικος

Είσοδος: 55  
Έξοδος: Ενήλικος

**Παρατήρηση:** Η δομή αυτή προσομοιώνει το **if..else** του προγραμματισμού

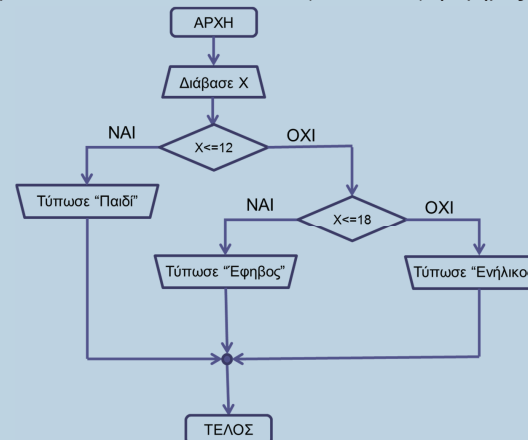
## A. Θεωρία

### 1. Διάγραμμα Ροής Προγράμματος

#### 5. Δομή Συνθήκης

Στον προγραμματισμό έχουμε συχνά περιπτώσεις που είναι περισσότερες από δύο. Αυτό μπορεί να προσομοιωθεί από ένα ΔΡΠ μέσω πολλαπλών ελέγχων.

**Παράδειγμα:** Το ακόλουθο ΔΡΠ προδιαγράφει έναν αλγόριθμο που διαβάζει την ηλικία ενός ατόμου και τυπώνει αν είναι παιδί ( $\leq 12$  ετών) ή έφηβος (13...18 ετών) ή ενήλικος ( $> 18$  ετών).



#### Παραδείγματα (εκτέλεσης):

Είσοδος: 9  
Έξοδος: Παιδί  
Είσοδος: 13  
Έξοδος: Έφηβος  
Είσοδος: 23  
Έξοδος: Ενήλικος

**Παρατήρηση:** Η δομή αυτή προσομοιώνει το **if...else if...else** του προγραμματισμού.

## A. Θεωρία

### 1. Διάγραμμα Ροής Προγράμματος

#### 5. Δομή Συνθήκης

**Άσκηση:** Κατασκευάστε Διάγραμμα Ροής Προγράμματος που ζητάει από το χρήστη να πληκτρολογήσει τον μισθό του και:

- Αν είναι μέχρι 500 ευρώ τυπώνει «Δεν έχει διακοπές»
- Αν είναι μεταξύ 500 και 1000 τυπώνει «Διακοπές στην Λούτσα»
- Αν είναι μεταξύ 1000 και 1500 τυπώνει «Διακοπές σε Νησί»
- Αν είναι παραπάνω από 1500 τυπώνει «Λες Ψέματα»

## A. Θεωρία

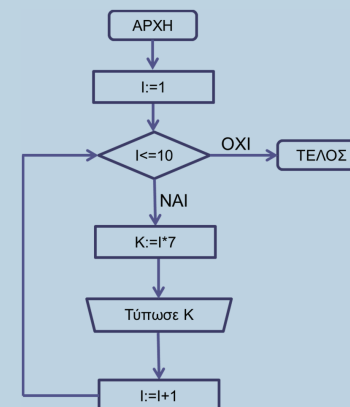
### 1. Διάγραμμα Ροής Προγράμματος

#### 6. Δομή Επανάληψης

Η επανάληψη είναι πολύ σημαντικό προγραμματιστικό εργαλείο. Επαναλαμβάνει πολλές φορές την εκτέλεση μιας ροής υπολογισμού όσο ικανοποιείται μια συνθήκη.

Στον προγραμματισμό υπάρχουν τρεις βασικές δομές επανάληψης για τις οποίες θα δούμε εδώ την υλοποίηση με ΔΡΠ.

**Παράδειγμα:** Να κατασκευαστεί ΔΡΠ που τυπώνει την προπαίδεια του 7.



#### Παράδειγμα

Έξοδος: 7, 14, 21, 28, 35, 42, 49  
56, 63, 70

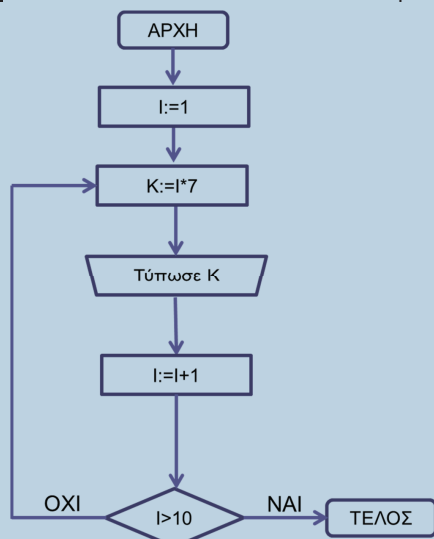
**Παρατήρηση:** Η δομή αυτή προσομοιώνει το **for** και το **while** του προγραμματισμού

## A. Θεωρία

### 1. Διάγραμμα Ροής Προγράμματος

#### 6. Δομή Επανάληψης

**Παράδειγμα 2:** Να κατασκευαστεί ΔΡΠ που τυπώνει την προπαίδεια του 7.



#### Παράδειγμα

Έξοδος: 7,14,21,28,35,42,49  
56,63,70

**Παρατήρηση:** Η δομή αυτή προσομοιώνει το **do..while** του προγραμματισμού

## A. Θεωρία

### 1. Διάγραμμα Ροής Προγράμματος

#### 6. Δομή Επανάληψης

Άσκηση 1: Κατασκευάστε Διάγραμμα Ροής Προγράμματος που υπολογίζει και τυπώνει το άθροισμα των αριθμών από το 1 έως το 10

## A. Θεωρία

### 1. Διάγραμμα Ροής Προγράμματος

#### 6. Δομή Επανάληψης

Άσκηση 2: Κατασκευάστε Διάγραμμα Ροής Προγράμματος που δέχεται ως είσοδο το N και τυπώνει το άθροισμα των τετραγώνων των αριθμών από το 1 έως το N.

## A. Θεωρία

### 1. Διάγραμμα Ροής Προγράμματος

#### 6.1. Δομή Επανάληψης (Εμφωλιασμένοι Βρόχοι)

**Παράδειγμα:** Το ΔΡΠ του σχήματος κάνει μια ενδιαφέρουσα εκτύπωση:

AAAAB AAAAB AAAAB

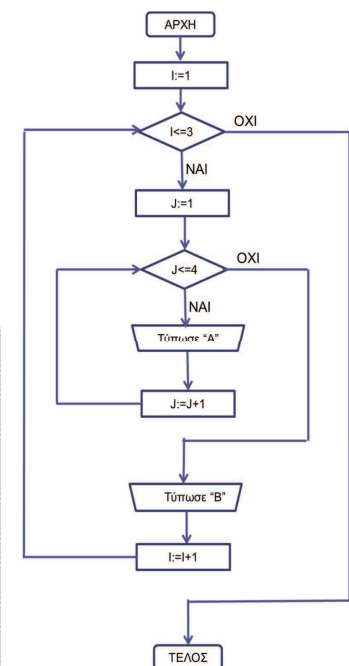
Συνεπώς εκτυπώνει 3 φορές τη συμβολοσειρά AAAAB υλοποιώντας μια διπλή επανάληψη

**Παρατήρηση:** Η δομή αυτή προσομοιώνει εμφωλιασμένους βρόχους μιας γλώσσας προγραμματισμού.

Συγκεκριμένα σε ψευδογλώσσα υλοποιεί το εξής τμήμα κώδικα:

```

ΕΠΑΝΕΛΑΒΕ από I:=1 έως 3
  ΕΠΑΝΕΛΑΒΕ από J:=1 έως 4
    Τύπωσε «A»
  ΤΕΛΟΣ-ΕΠΑΝΕΛΑΒΕ
  Τύπωσε «B»
ΤΕΛΟΣ-ΕΠΑΝΕΛΑΒΕ
  
```





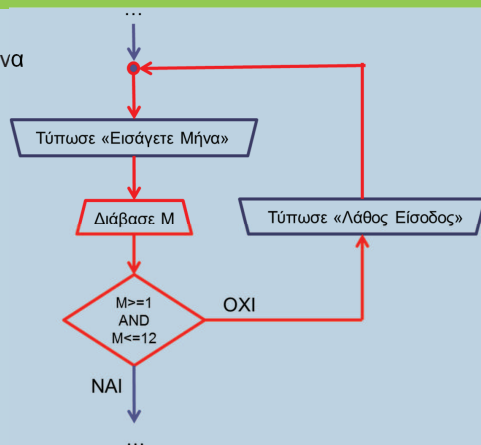
## Α. Θεωρία

### 1. Διάγραμμα Ροής Προγράμματος

#### 6.2. Αμυντικός Προγραμματισμός

Ο αμυντικός προγραμματισμός είναι μια προγραμματιστική τεχνική με την οποία ελέγχεται η εγκυρότητα των δεδομένων εισόδου. Δηλαδή όταν ο χρήστης πληκτρολογεί την είσοδο, ελέγχεται αν αυτή είναι έγκυρη. Αν είναι λάθος, τότε του ζητείται να την επαναπληκτρολογήσει. Αν είναι ορθή, τότε προχωράμε στο κυρίως πρόγραμμα.

**Παράδειγμα:** Το ακόλουθο τμήμα ΔΡΠ ελέγχει αν η είσοδος του χρήστη αντιστοιχεί σε ένα μήνα (άρα θα πρέπει να έχει τιμή από 1 έως 12)



#### Παράδειγμα Εκτέλεσης

Εισάγετε μήνα: 17

Λάθος Είσοδος

Εισάγετε μήνα 11

...



## Α. Θεωρία

### 1. Διάγραμμα Ροής Προγράμματος

#### 7. Πίνακες

Ένας πίνακας είναι μία δομή δεδομένων που ενοποιεί πολλές μεταβλητές ώστε να χρησιμοποιούνται με έναν εύκολο τρόπο.

- Οι μεταβλητές ενός μονοδιάστατου πίνακα απεικονίζονται:

$A[1], A[2], \dots, A[N]$

- Οι μεταβλητές ενός διδιάστατου πίνακα απεικονίζονται:

$A[1,1], A[1,2], \dots, A[1,N]$

$A[2,1], A[2,2], \dots, A[2,N]$

...

$A[M,1], A[M,2], \dots, A[M,N]$

Η διαπέραση των πινάκων γίνεται με δομή επανάληψης (με απλό βρόχο των μονοδιάστατων πινάκων και με διπλό βρόχο των διδιάστατων πινάκων).

Αυτό είναι ιδιαίτερα σημαντικό προγραμματιστικό στοιχείο και θα δούμε στις επόμενες δύο ασκήσεις την χρήση του.



## Α. Θεωρία

### 1. Διάγραμμα Ροής Προγράμματος

#### 6.2. Αμυντικός Προγραμματισμός

Άσκηση: Κατασκευάστε Διάγραμμα Ροής Προγράμματος που λαμβάνει ως είσοδο δύο αριθμούς από το 1 έως το 10 (εφαρμόστε αμυντικό προγραμματισμό και έπειτα τυπώνει το άθροισμα των τετραγώνων τους).



## Α. Θεωρία

### 1. Διάγραμμα Ροής Προγράμματος

#### 7. Πίνακες

Άσκηση: Κατασκευάστε Διάγραμμα Ροής Προγράμματος που διαβάζει από τον χρήστη έναν αριθμό  $N$ , έπειτα αποθηκεύει στην  $i$ -οστή θέση του πίνακα τον αριθμό  $1 / i^2$  και έπειτα υπολογίζει το άθροισμα όλων των στοιχείων του πίνακα



## A. Θεωρία

### 1. Διάγραμμα Ροής Προγράμματος

#### 7. Πίνακες

Άσκηση: Κατασκευάστε Διάγραμμα Ροής Προγράμματος που διαβάζει το M και το N, έπειτα διαβάζει τα MxN στοιχεία του πίνακα και έπειτα υπολογίζει και τυπώνει το άθροισμά τους.