

**Γυμνάσιο Αρχαγγέλου**  
**Σχ. έτος: 2010 - 2011**  
**Εξεταστέα ύλη περιόδου Μαΐου - Ιουνίου**  
**Τάξη: Γ**  
**Μάθημα: Πληροφορική**

**Θεωρία**

Παράγραφοι: 1.1, 1.2, 1.3 (σελίδες: 176 έως 181)

**Προγραμματισμός σε LOGO**

**-> Εντολές σχεδίασης**

| <b>Όνομα εντολής</b> | <b>Περιγραφή</b>  | <b>παράδειγμα</b>                           |
|----------------------|---|---|
| fd (forward)         | Προχωράει η χελώνα μπροστά  | fd 100 (προχωράει μπροστά 100 μονάδες)      |
| rt (right turn)      | Στρίβει η χελώνα δεξιά  | rt 45 (στρίβει δεξιά 45 μοίρες)             |
| lt (left turn)       | Στρίβει η χελώνα αριστερά   | lt 60 (στρίβει αριστερά 60 μοίρες)          |
| bk (back)            | Προχωράει η χελώνα προς τα πίσω (με την όπισθεν)                                | bk 80 (προχωράει με την όπισθεν 80 μοναδες) |
| pu (pen up)          | Σηκώνεται το στυλό, με αποτέλεσμα όταν κινείται η χελώνα να μη σχεδιάζει γραμμή | -   |
| pd (pen down)        | Κατεβαίνει το στυλό κάτω, με αποτέλεσμα όταν κινείται η χελώνα να σχεδιάζει     | -   |

**-> Μεταβλητές στη Logo**

Για να καταχωρίσω τιμή σε μία μεταβλητή, χρησιμοποιώ την εντολή **make**.

Όταν θέλω **να καταχωρίσω** τιμή σε μία μεταβλητή, τη γράφω με **διπλά εισαγωγικά** ( " ) στην αρχή της.

Παραδείγματα:

- make "x 10 (καταχωρεί στη μεταβλητή x την τιμή 10, οπότε η x γίνεται ίση με τον αριθμό 10)

- `make "y 20` (καταχωρεί στη μεταβλητή  $y$  την τιμή 20, οπότε η  $y$  γίνεται ίση με τον αριθμό 20)
- `make "x (3+4)` (καταχωρεί στη μεταβλητή  $x$  το άθροισμα  $3+4$ , οπότε η  $x$  γίνεται ίση με τον αριθμό 7)
- `make "z (2*5)` (καταχωρεί στη μεταβλητή  $z$  το γινόμενο  $2*5$ , οπότε η  $z$  γίνεται ίση με τον αριθμό 10)

Όταν θέλω **να χρησιμοποιήσω** μία μεταβλητή που της έχω ήδη δώσει τιμή, τη γράφω με **άνω-κάτω τελεία** ( : ) στην αρχή της. Για παράδειγμα, αν γράψω τις παρακάτω εντολές:

```
make "x 10
label :x
```

θα μου εμφανίσει την τιμή της μεταβλητής  $x$ , που είναι το 10.

Μπορώ να χρησιμοποιήσω μεταβλητές για να δώσω τιμές σε άλλες μεταβλητές. Για παράδειγμα, εάν γράψω τις παρακάτω εντολές:

```
make "a 10
make "b 20
make "z (:a + :b)
make "w (:a * :b)
make "k (:z + :w)
```

οι μεταβλητές θα έχουν στο τέλος τις παρακάτω τιμές:

| Όνομα μεταβλητής | Πράξη      | Τιμή μεταβλητής |
|------------------|------------|-----------------|
| a                | -          | 10              |
| b                | -          | 20              |
| z                | $10 + 20$  | 30              |
| w                | $10 * 20$  | 200             |
| k                | $30 + 200$ | 230             |

### ->Η εντολή **repeat**

Με την εντολή **repeat**, μπορώ να εκτελέσω εντολές **που επαναλαμβάνονται πολλές φορές**.

Για παράδειγμα, για να σχεδιάσω ένα τετράγωνο με πλευρά 100, χρησιμοποιώ τις παρακάτω εντολές:

```
fd 100
rt 90
fd 100
rt 90
```

```
fd 100
rt 90
fd 100
```

Παρατηρώ ότι οι εντολές `fd 100` μαζί με την `rt 90` επαναλαμβάνονται 4 φορές. Με τη χρήση της `repeat`, μπορώ να σχεδιάσω το τετράγωνο γράφοντας μόνο την παρακάτω γραμμή:

```
repeat 4 [fd 100 rt 90]
```

Ο αριθμός μετά την `repeat` (το 4 στο παράδειγμά μας) καθορίζει πόσες φορές θα εκτελεστούν οι εντολές που υπάρχουν μέσα στις αγκύλες.

Η εντολή:

```
repeat 6 [fd 100 rt 60]
```

σχεδιάζει ένα εξάγωνο με πλευρά 100 μονάδες. Η χελώνα δηλαδή, θα κινηθεί μπροστά 100 μονάδες (`fd 100`) και θα στρίψει δεξιά 60 μοίρες (`rt 60`). Αυτό θα επαναληφθεί συνολικά 6 φορές, σχεδιάζοντας τελικά ένα εξάγωνο.

Μπορούμε να συνδιάσουμε τη `repeat` με μεταβλητές. Για παράδειγμα οι εντολές:

```
make "x 100
repeat 4 [fd :x rt 90]
```

σχεδιάζουν ένα τετράγωνο με πλευρά 100 (το `x` είναι ίσο με 100).

Ενώ οι εντολές

```
make "x 80
repeat 4 [fd :x rt 90]
```

σχεδιάζουν ένα τετράγωνο με πλευρά 80 (το `x` είναι ίσο με 80).

### ->Ομαδοποίηση εντολών

Πολλές φορές είναι πολύ χρήσιμο να ομαδοποιήσω εντολές και μάλιστα μπορώ να δώσω και όνομα στην ομάδα που δημιουργώ. Για παράδειγμα, για να σχεδιάσω ένα τετράγωνο, μπορώ να δημιουργήσω την ομάδα **tetragono** ως εξής:

```
to tetragono
repeat 4 [fd 100 rt 90]
end
```

Πριν ορίσω την ομάδα, ξεκινάω με τη λέξη **to**, ακολουθεί το όνομα της ομάδας

(tetragono στο παράδειγμά μας), ακολουθούν οι εντολές και η ομάδα τελειώνει με τη λέξη **end**.

Σε ένα άλλο παραδειγμα, δημιουργώ μια ομάδα με το όνομα **eksagono** που σχεδιάζει ένα εξάγωνο, αφού πρώτα στρίψει η χελώνα δεξιά 90 μοίρες:

```
to eksagono
rt 90
repeat 6 [fd 100 rt 60]
end
```

Αν τώρα γράψω απλά **eksagono**, θα στρίψει η χελώνα δεξιά 90 μοίρες και θα σχεδιάσει το εξάγωνο.

Μπορώ επίσης να χρησιμοποιήσω ομάδες μέσα σε ομάδες. Για παράδειγμα:

```
to 2_tetragona
tetragono
lt 90
fd 50
tetragono
end
```

Αν γράψω τώρα **2\_tetragona**, θα σχεδιάσει η χελώνα ένα τετράγωνο, θα στρίψει αριστερά 90 μοίρες, θα προχωρήσει μπροστά 50 μονάδες και θα σχεδιάσει άλλο ένα τετράγωνο.

Μπορώ να χρησιμοποιήσω τις ομάδες που έχω φτιάξει και μέσα σε **repeat**. Για παράδειγμα, αν γράψω:

```
repeat 50 [tetragono rt 60]
```

η χελώνα θα σχεδιάσει ένα τετράγωνο και θα στρίψει 60 μοίρες δεξιά, και θα πραγματοποιήσει τις δύο αυτές κινήσεις 50 φορές.

**Οι διδάσκοντες**

Πομπός Σάββας

Αλατζάς Χρήστος