

Αντικειμενοστρεφής Προγραμματισμός - Python

Κ.Π. Γιαλούρης

Στόχοι του σημερινού μαθήματος

- ❑ Κατανόηση της αναγκαιότητας της επανάληψης σε ένα πρόγραμμα.
- ❑ Παρουσίαση των εντολών επανάληψης στη **Python** καθώς και των βασικών χαρακτηριστικών τους.
- ❑ Κατανόηση της εντολής επανάληψης **for**.
- ❑ Χρήση της εντολής **for** σε απλά και σύνθετα προβλήματα.

Εντολές Επανάληψης

- ❑ Εντολές επανάληψης που οι επαναλήψεις τερματίζουν μετά από ένα προκαθορισμένο αριθμό
- ❑ Εντολές επανάληψης που οι επαναλήψεις τερματίζουν μετά το έλεγχο μίας συνθήκης

Εντολή Επανάληψης - `for`

- ❑ Η εντολή `for` είναι μια απλή δομή επανάληψης που επαναλαμβάνει ένα block εντολών ένα συγκεκριμένο πλήθος επαναλήψεων.
- ❑ Η εντολή `for...in...` είναι μία εντολή βρόχου, η οποία είναι σχεδιασμένη να λειτουργεί τόσες φορές όσες προκύπτει από μία αλληλουχία δεδομένων.

Εντολή Επανάληψης - `for`

- ❑ Η αλληλουχία_δεδομένων είναι ένα σύνολο διακριτών δεδομένων που προκύπτουν από:
 - ❑ Μία συνάρτηση που ορίζει ένα κλειστό διάστημα ακεραίων
 - ❑ Τους χαρακτήρες ενός `string`, τα στοιχεία μίας λίστας (`list`), μίας πλειάδας(`tuple`), ενός λεξικού (`dictionary`), ενός συνόλου(`set`), ή μιας πιο πολύπλοκης δομής.

Σύνταξη εντολής **for**

```
for μεταβλητή_for in αλληλουχία_δεδομένων :  
    εντολή-1  
    εντολή-2  
    .....  
    εντολή-ν  
επομενη_μετα_for_εντολή
```

Η μεταβλητή **for** είναι το όνομα μιας μεταβλητής η οποία θα πάρει τιμές μέσα από τη δομή που ονομάσθηκε **αλληλουχία_δεδομένων**

Η συνάρτηση `range`

Η συνάρτηση `range` παρέχει στην εντολή `for` παραμέτρους για την εκτέλεση ενός `block` επανάληψης. Η σύνταξη και οι λειτουργίες της `range` είναι :

Η συνάρτηση `range`

`range(n)` όπου `n` ακέραιος αριθμός

Λειτουργία: Δημιουργεί την αλληλουχία διαδοχικών αριθμών `0,1,2,3,4,...,n-1`

`range(from, to)` όπου `from, to` ακέραιοι

Λειτουργία: Δημιουργεί την αλληλουχία διαδοχικών αριθμών στο διάστημα `[from, to-1]`

`range(from, to, step)` όπου `from, to, step` ακέραιοι

Λειτουργία: Δημιουργεί την αλληλουχία αριθμών στο διάστημα `[from, to-step]` ως εξής :

`from, from+step, from+step+step,, to-step]`

Η συνάρτηση range

□ Αν $to > from$ τότε πρέπει $step > 0$

■ `Range[from, to, step]`

`[from, from+step, from+step+step,, to-step]`

□ Αν $to < from$ τότε πρέπει $step < 0$

■ `Range[from, to, step]`

`[from, from-step, from-step-step,, to+step]`

Η εντολή for με χρήση λίστας

```
lista=[timi1,timi2,...timin]
```

```
for i in lista:
```

```
    entoli_1
```

```
    entoli_2
```

```
    .....
```

```
    entoli_n
```

Η εντολή for με χρήση λίστας

```
lista=[1,4,5,7,8,9]
for i in lista:
    print(i)
```

- ❑ Η λίστα αποτελείται από 6 στοιχεία
- ❑ Η επανάληψη εκτελείται 6 φορές
- ❑ Σε κάθε κύκλο επανάληψης η τιμή της μεταβλητής παίρνει ως τιμή την αντίστοιχη τιμή της λίστας

Η εντολή for με χρήση λίστας

```
lista=[1,4,5,7,8,9]
for i in lista:
    print(i)
```

Το αποτέλεσμα θα είναι:

```
1
4
5
7
8
9
```

Η εντολή for με χρήση range

```
for i in range(4):  
    print(i)
```

□ Το αποτέλεσμα θα είναι:

```
0  
1  
2  
3
```

Η εντολή for με χρήση range

```
for i in range(2, 6):  
    print(i)
```

□ Το αποτέλεσμα θα είναι:

2
3
4
5

Η εντολή for με χρήση range

```
for i in range(0,10,2):  
    print(i)
```

□ Το αποτέλεσμα θα είναι:

0
2
4
6
8

Η εντολή for με χρήση range

```
for i in range(10, 0, -2):  
    print(i)
```

□ Το αποτέλεσμα θα είναι:

```
10  
8  
6  
4  
2
```


Εντολή Επανάληψης `While`

Η εντολή `while` είναι μία εντολή επανάληψης στην οποία ο έλεγχος της επανάληψης πραγματοποιείται μέσω μίας λογικής πρότασης.

Εντολή Επανάληψης `while`

Η δομή της εντολή είναι:

`while` *λογική_προταση:*

εντολή-1

εντολή-2

εντολή-3

.....

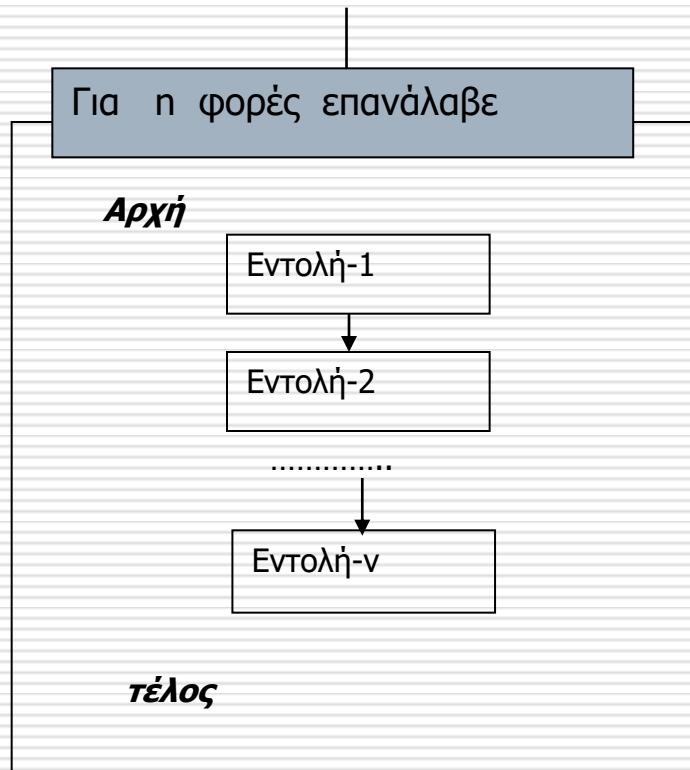
εντολή-ν

επόμενη εντολή

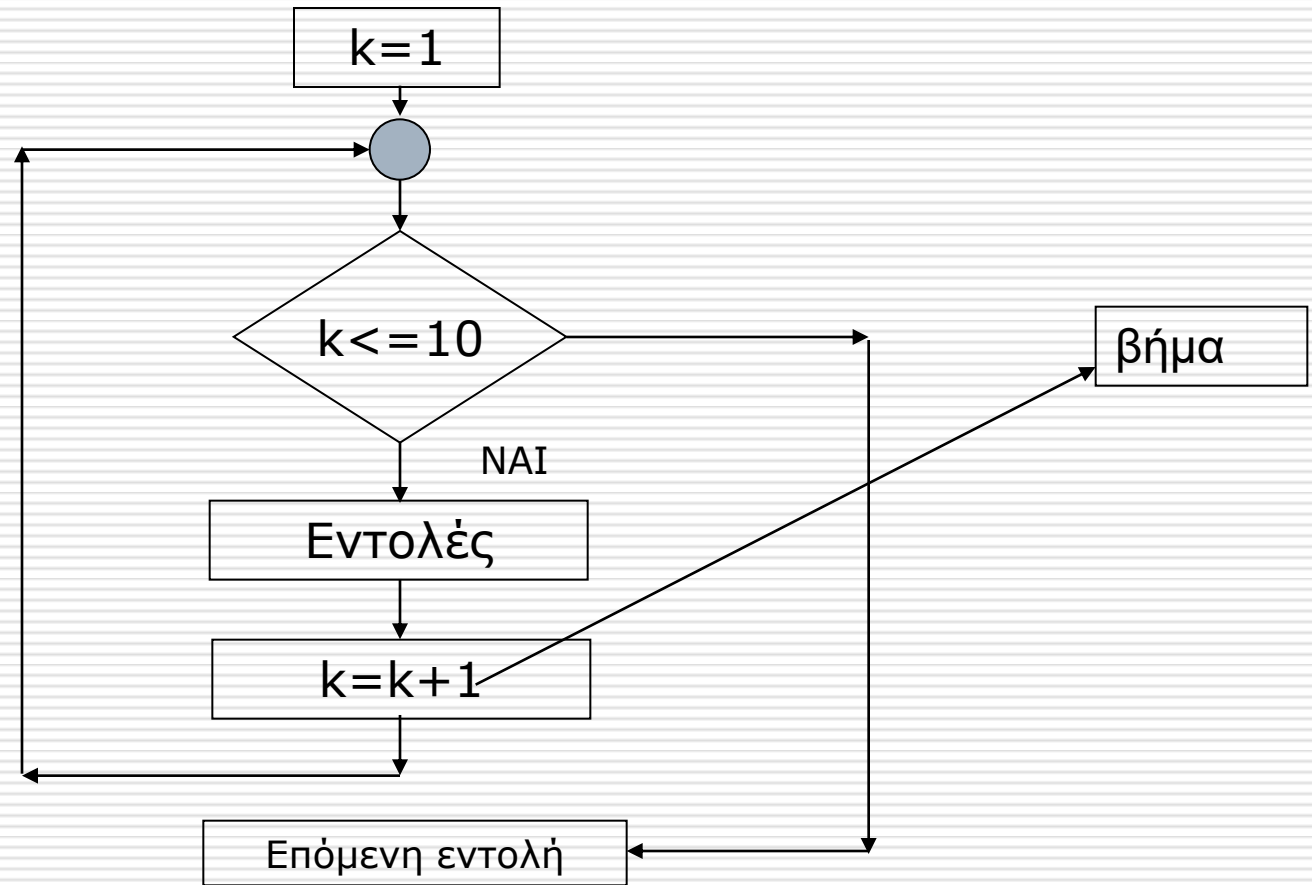
Εντολή Επανάληψης While

- ❑ Σε αυτή τη μορφή επανάληψης, οι εντολές που βρίσκονται στο block επανάληψης θα εκτελεστούν για μη προκαθορισμένο αριθμό επαναλήψεων.
- ❑ Η επανάληψη τερματίζει με βάση την τιμή της λογικής πρότασης.
- ❑ Με το όρο **λογική_πρόταση** θεωρείται ότι μπορεί να είναι:
 - Μία μεταβλητή που μπορεί να πάρει τιμές **True** ή **False** ή
 - μία απλή συνθήκη με χρήση σχεσιακών τελεστών ή
 - μία σύνθετη συνθήκη με χρήση λογικών τελεστών ή.
 - η σταθερά **True**.

For ...



For ...



Άσκηση -1

- Να γραφεί πρόγραμμα το οποίο
 - Εμφανίζει στην οθόνη το άθροισμα των αριθμών από 1 έως και 100.

Άσκηση -2

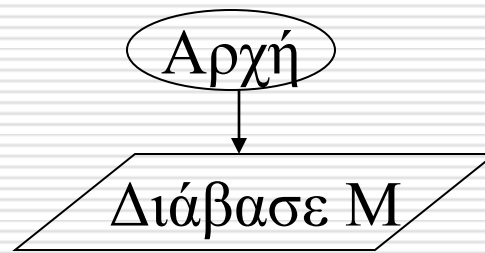
- Να γραφεί πρόγραμμα το οποίο
 - Εμφανίζει στην οθόνη το άθροισμα των περιττών αριθμών από 1 έως και 100.

Άσκηση -3

- Να γραφεί πρόγραμμα το οποίο να διαβάζει από την οθόνη 5 αριθμούς και στο τέλος να εμφανίζει τον μεγαλύτερο.

Αρχή





Αρχή



Διάβασε M



MAX ←= M



Αρχή

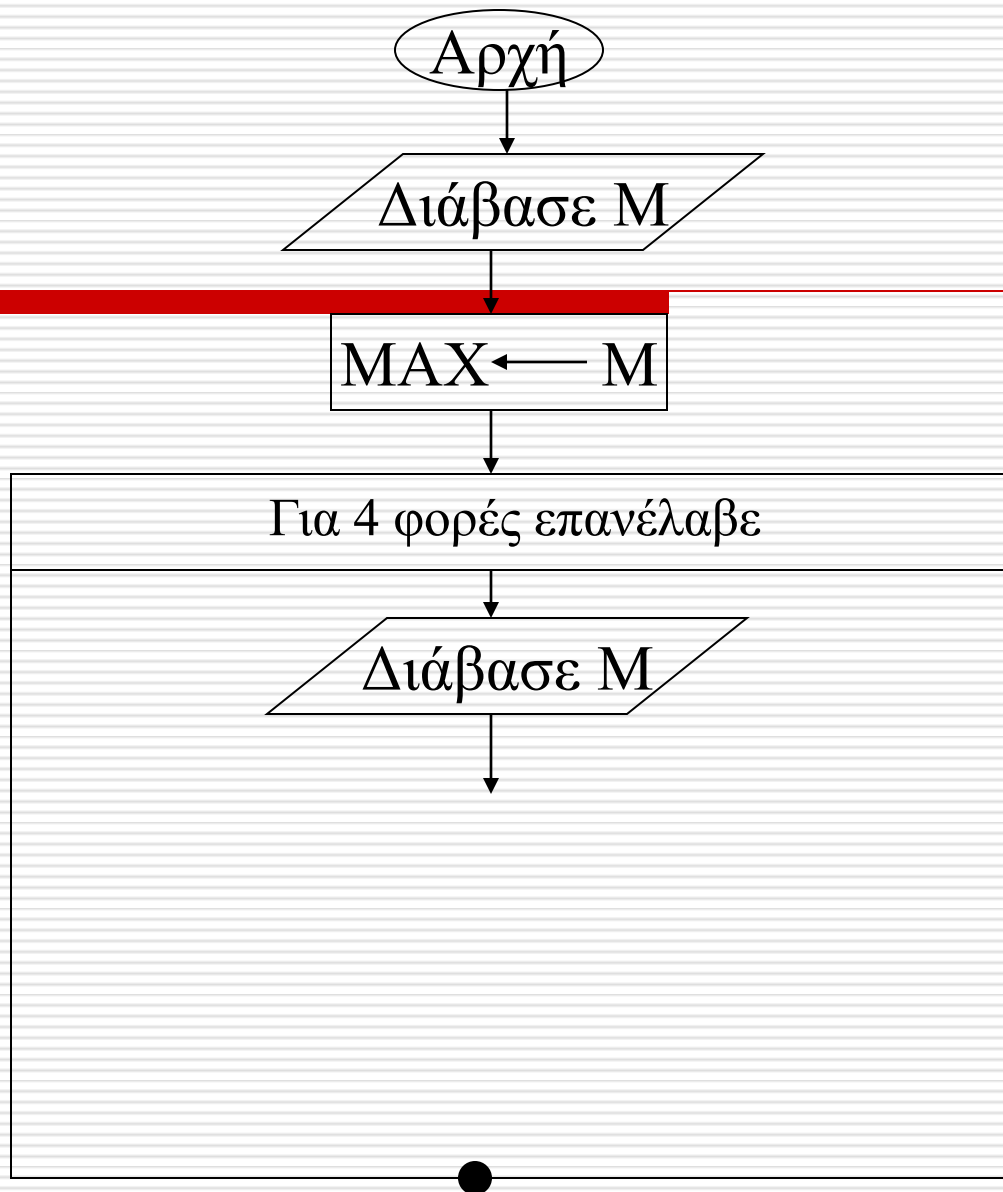


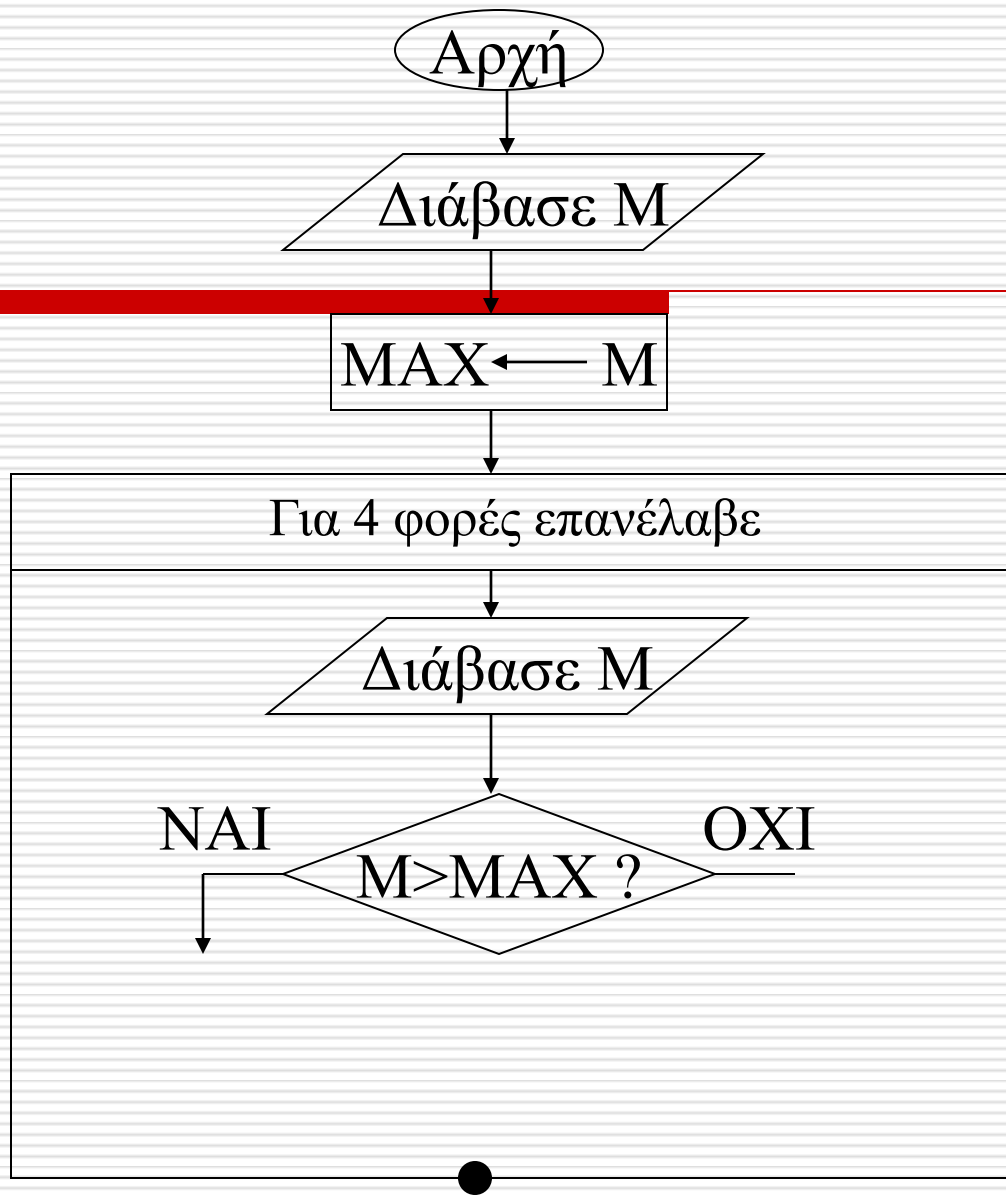
Διάβασε M

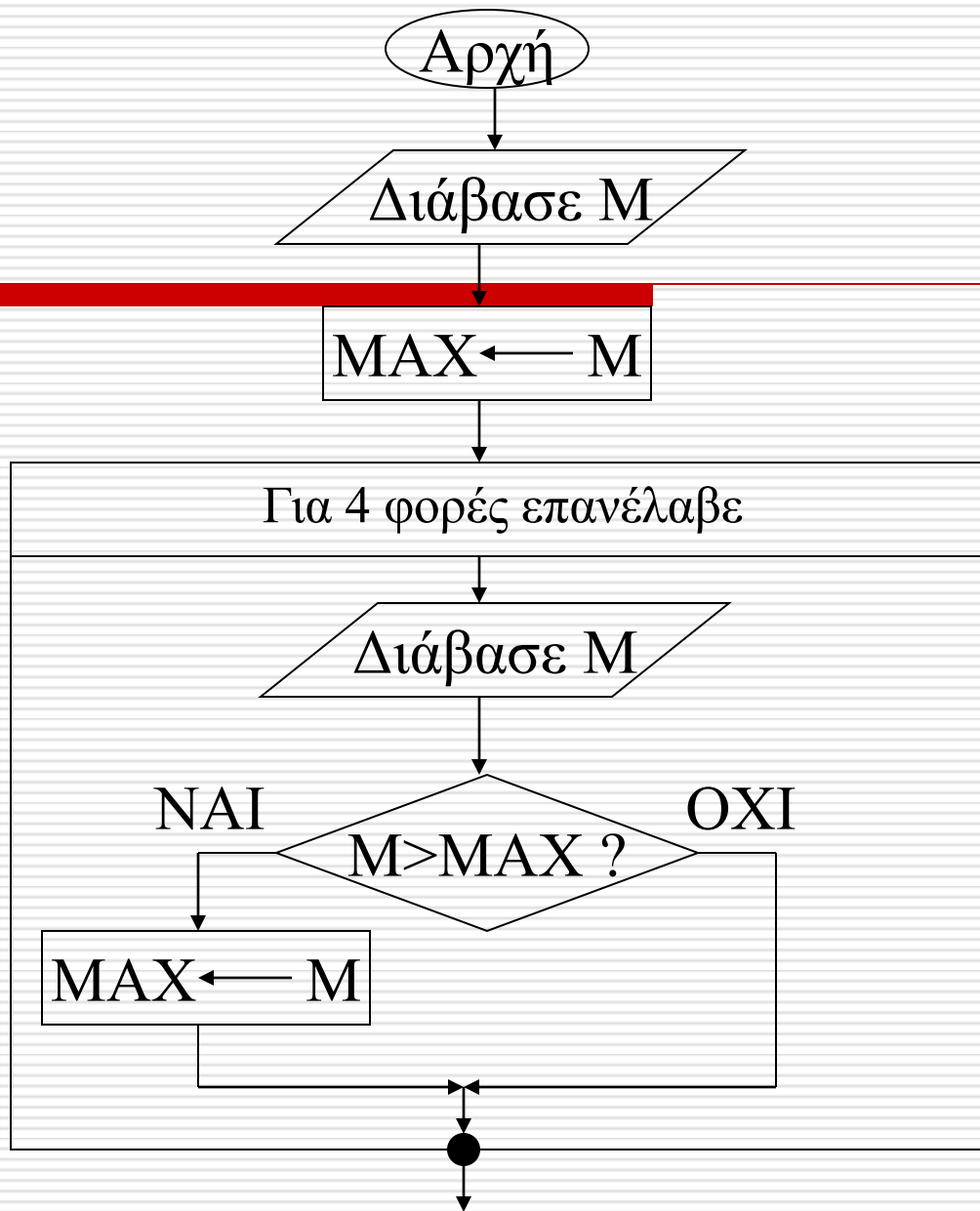


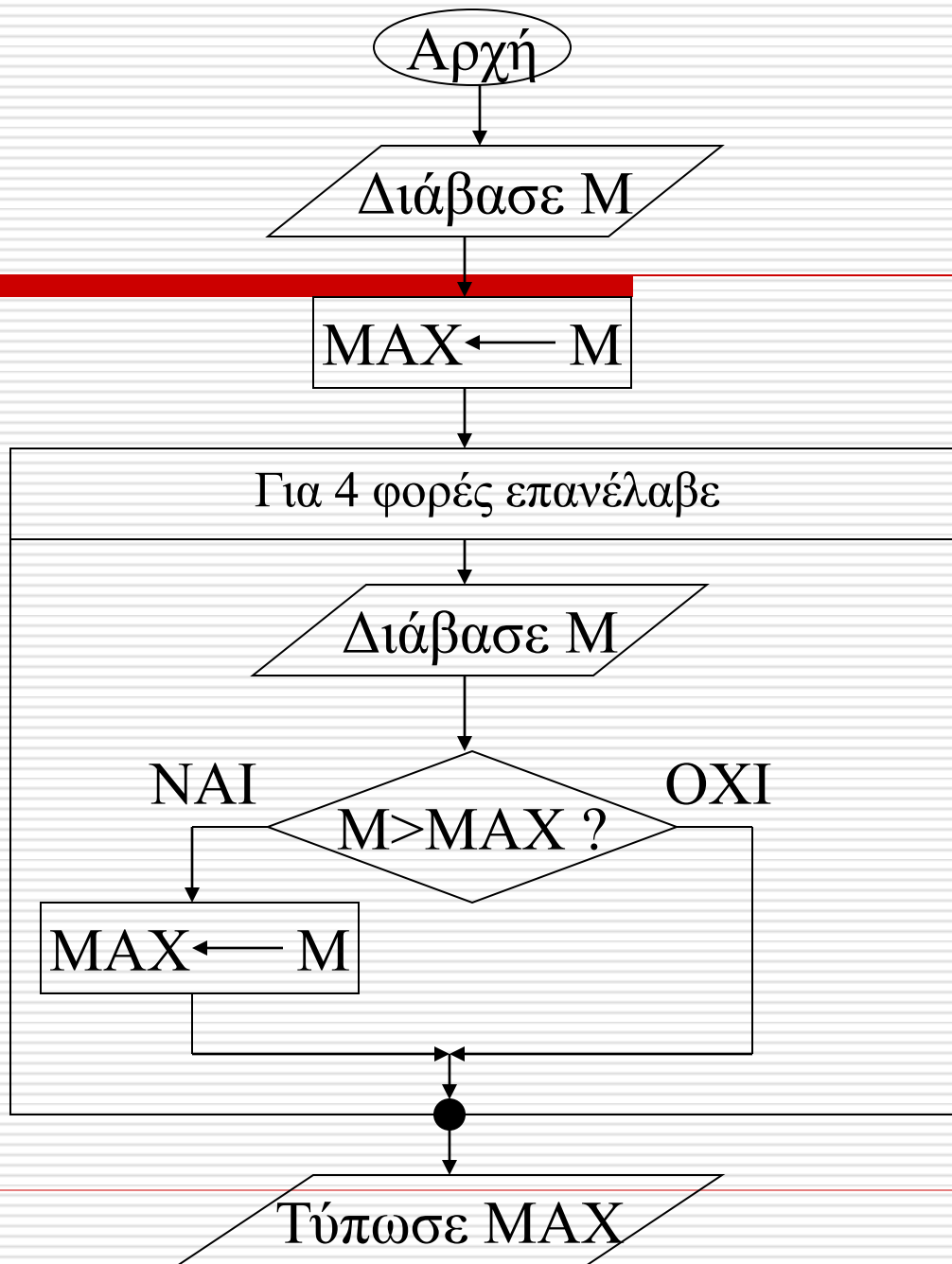
$MAX \leftarrow M$

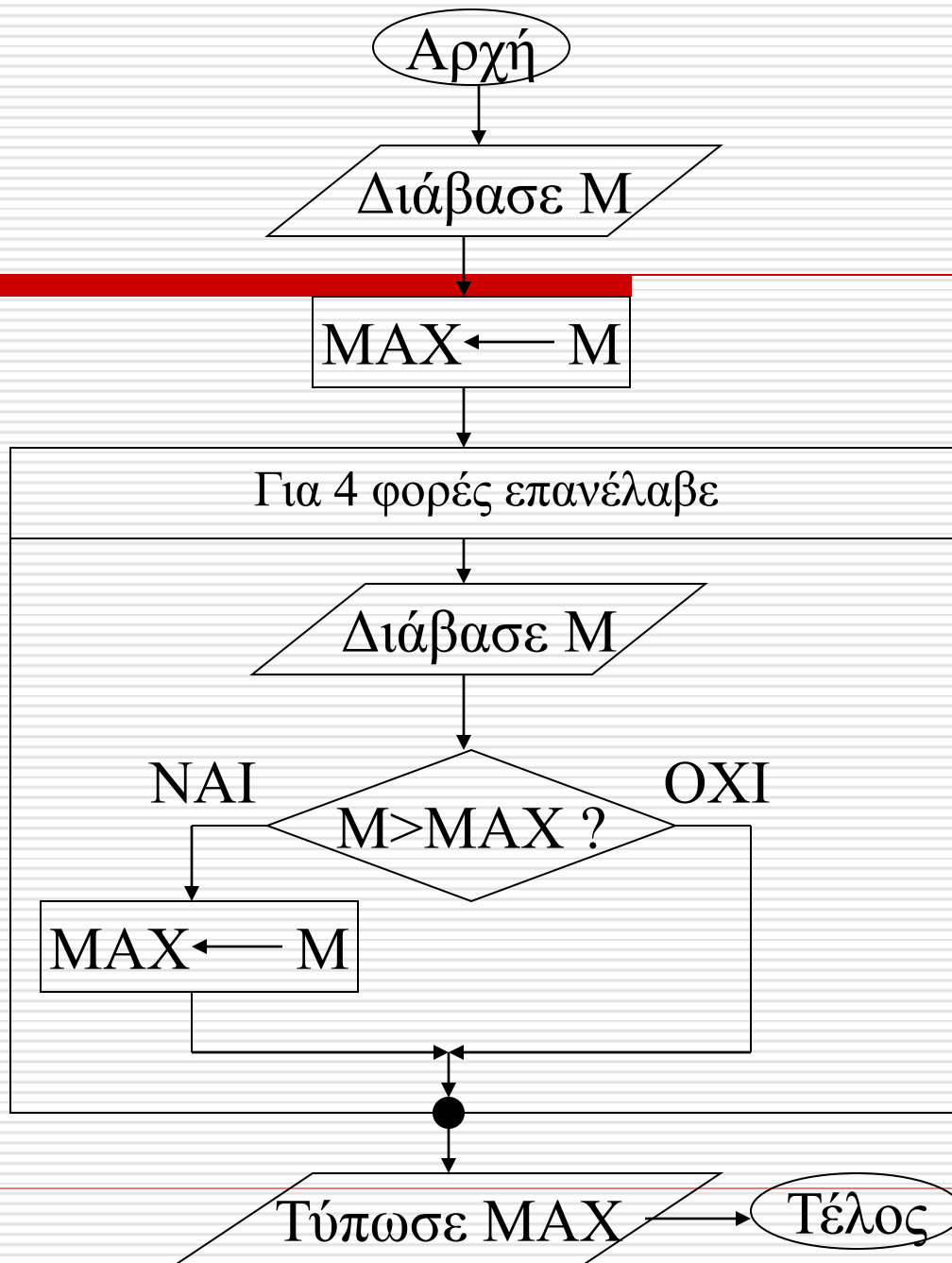


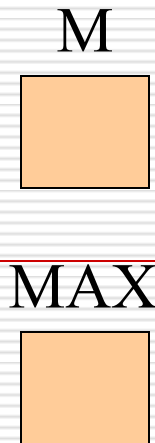
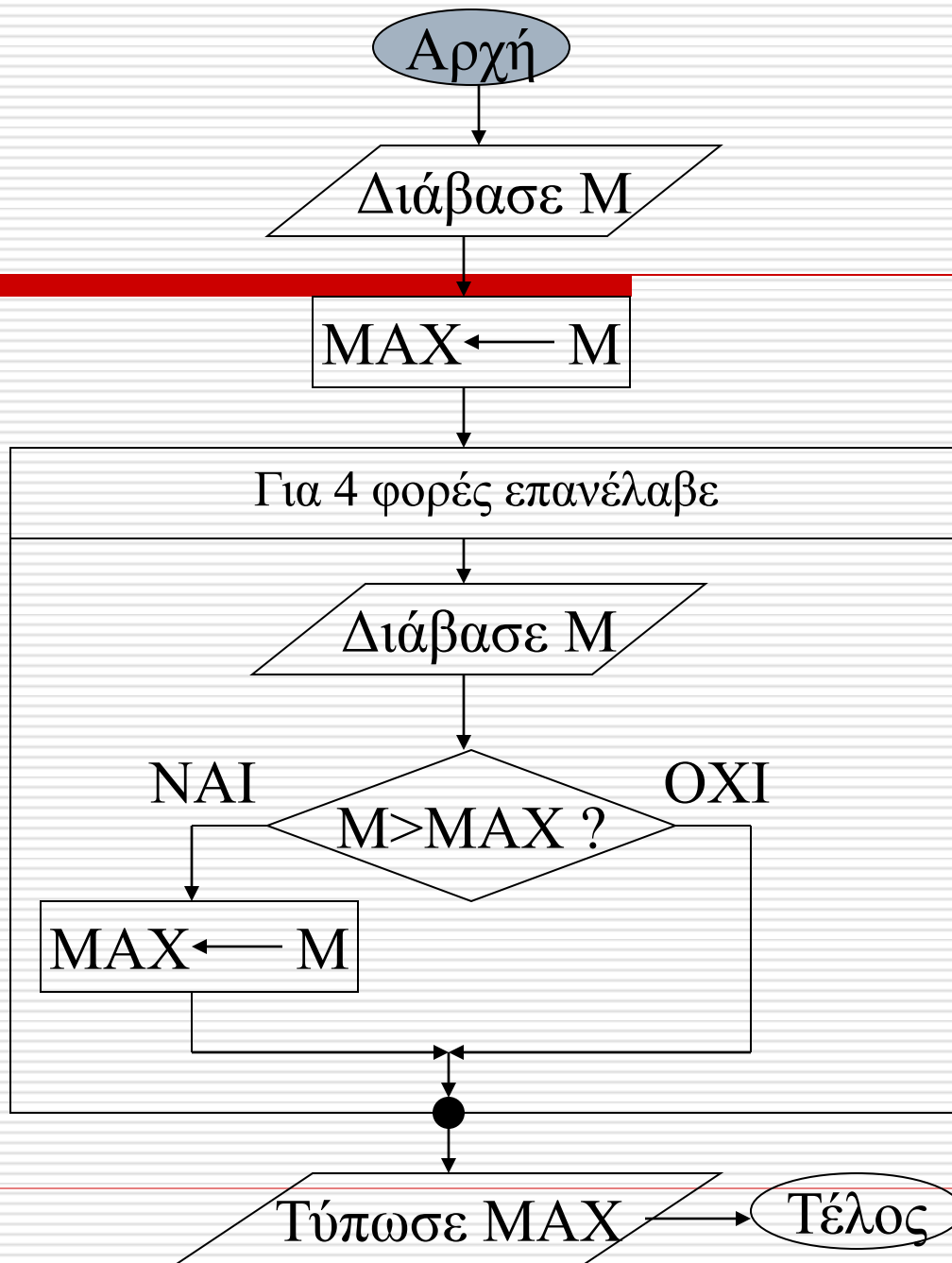
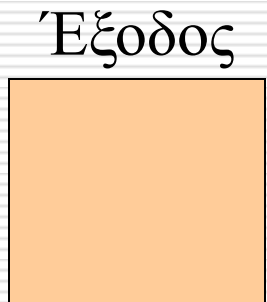
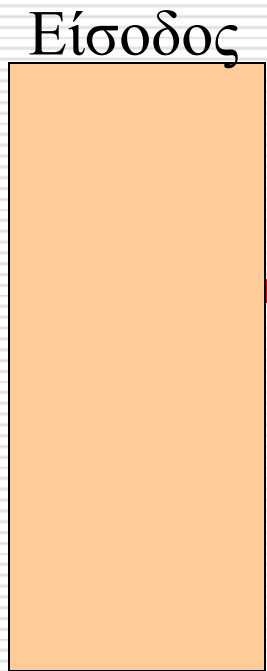


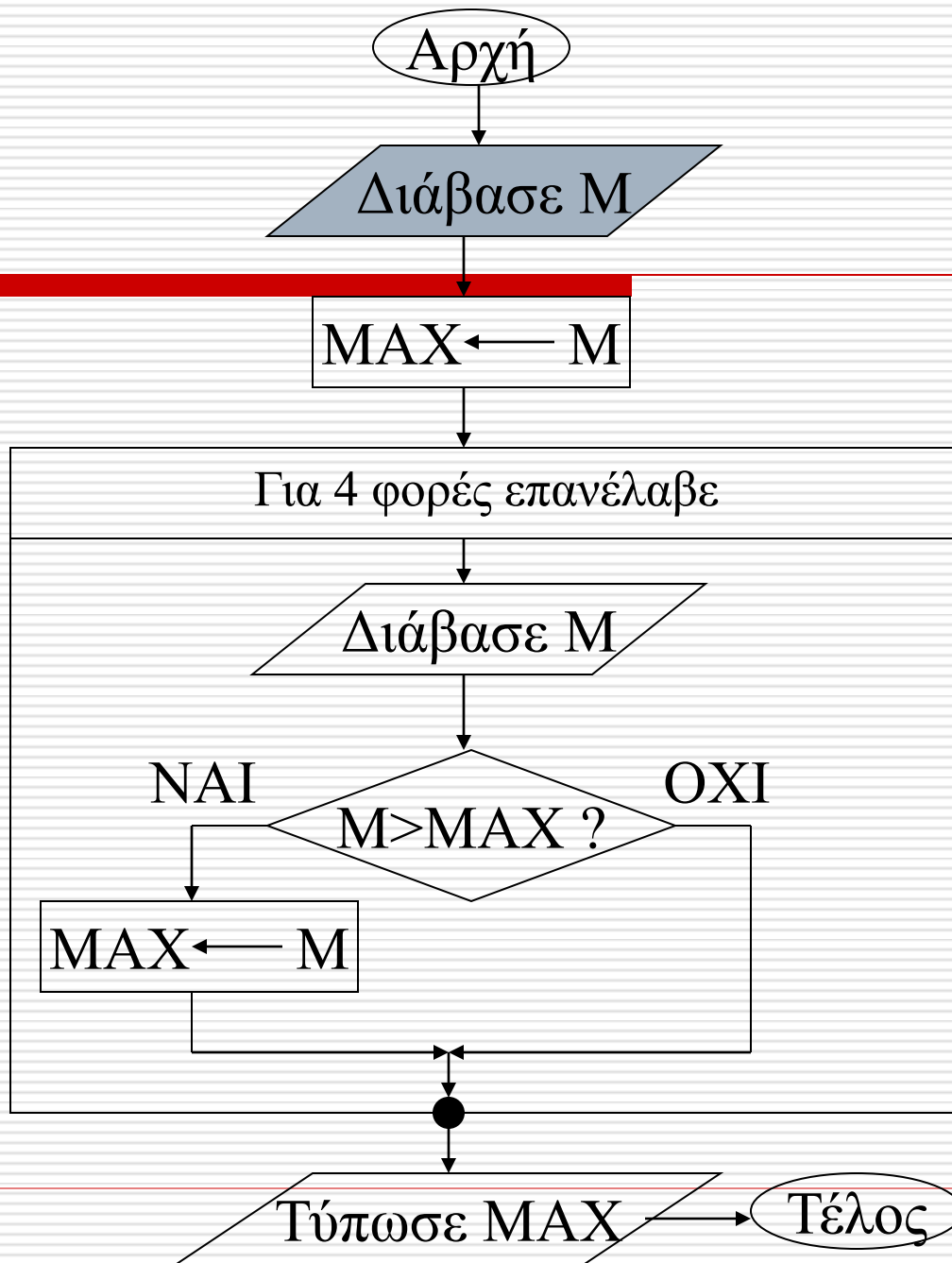
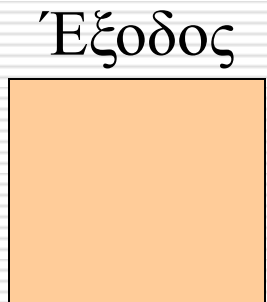
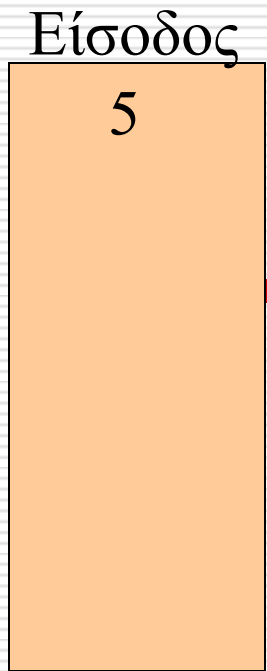


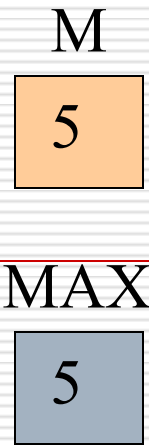
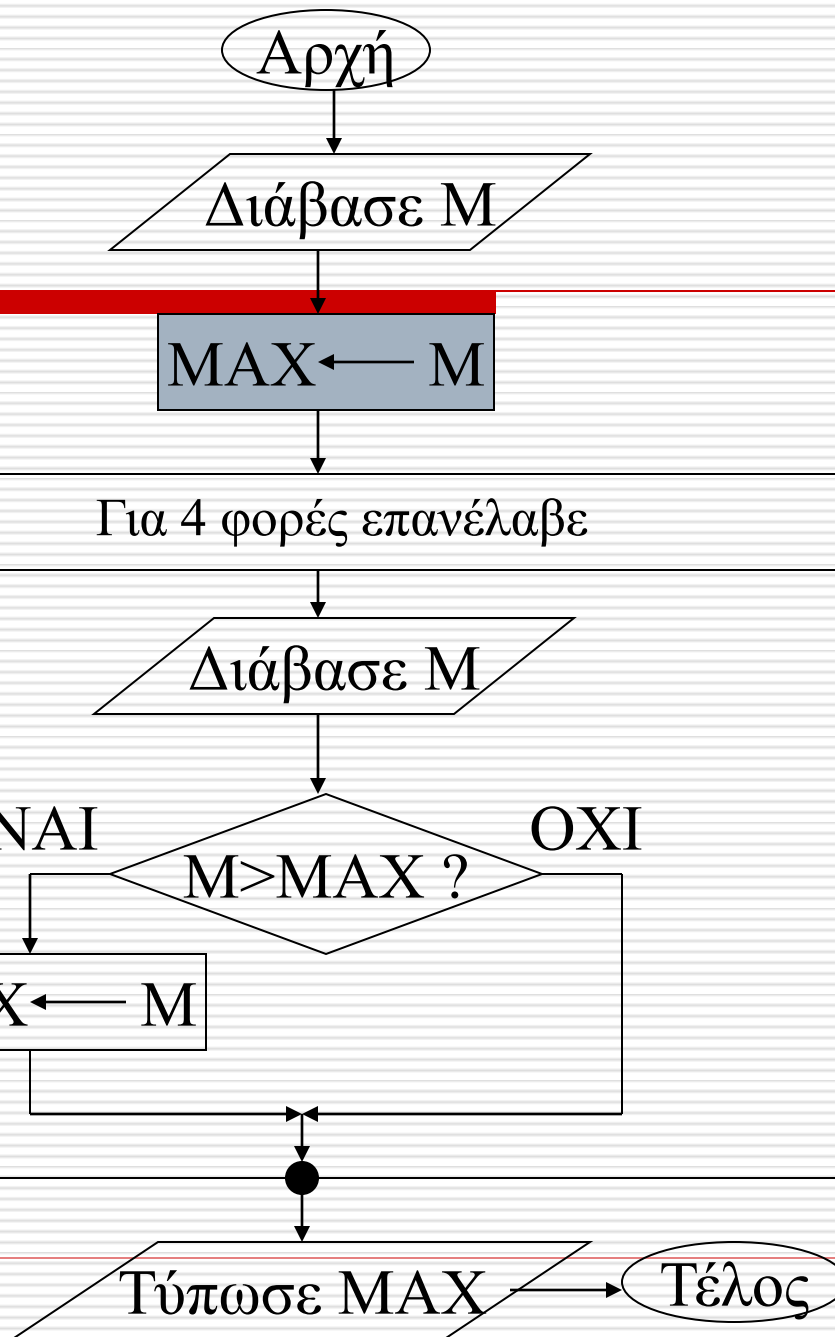
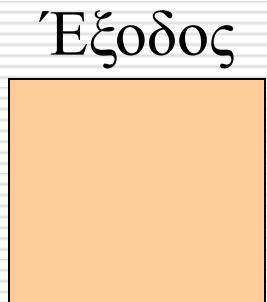
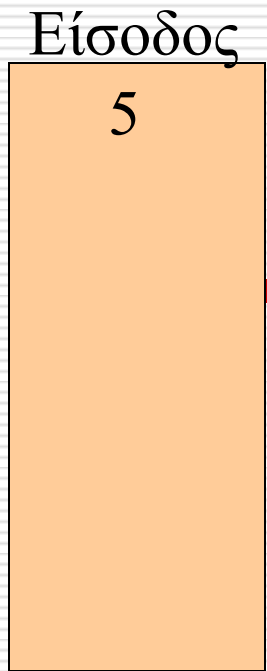


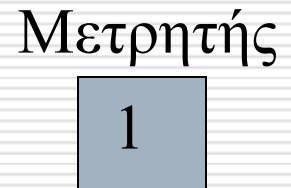
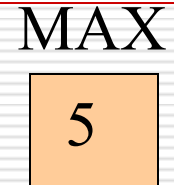
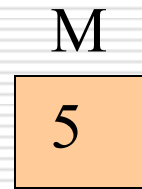
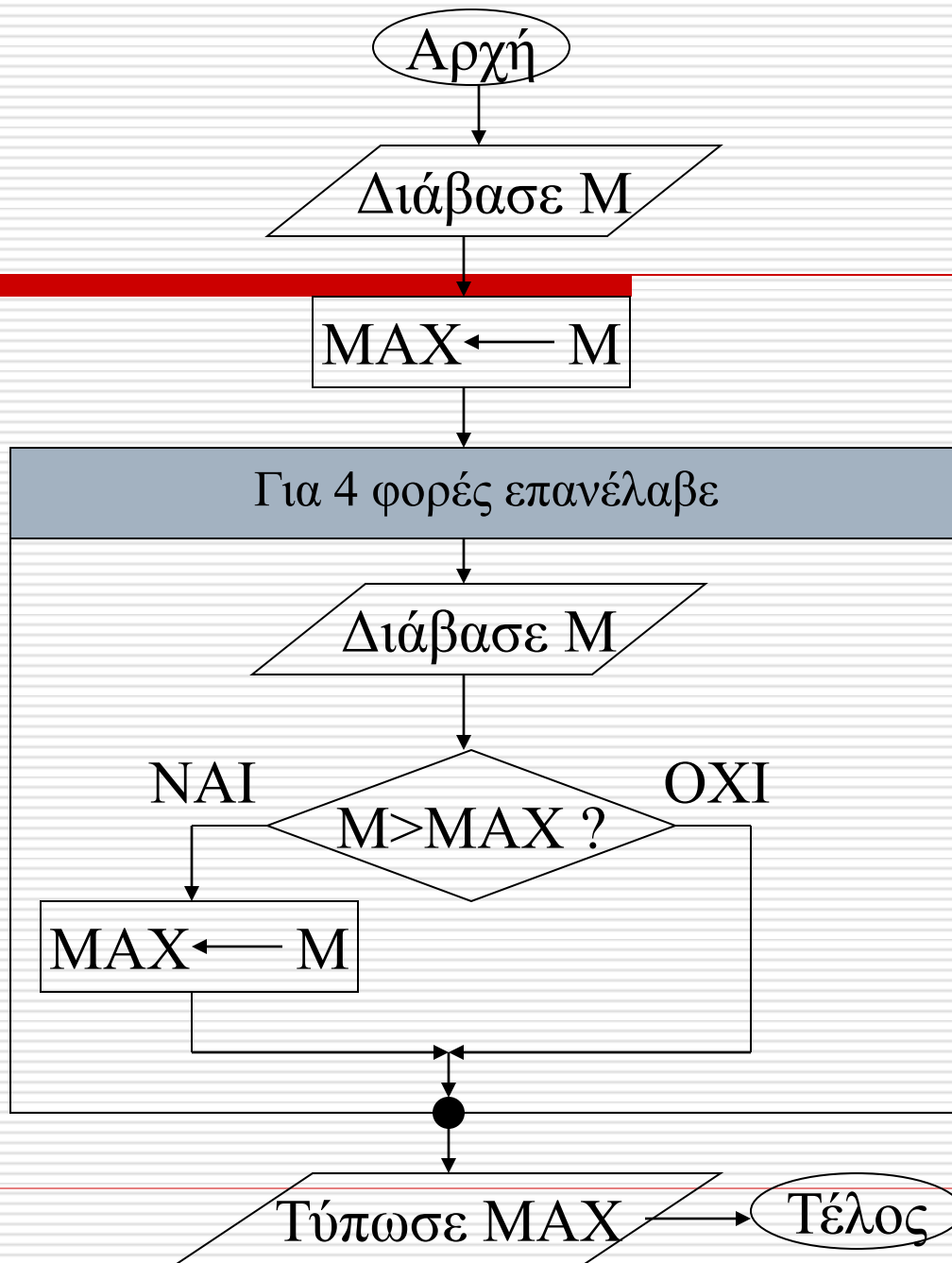
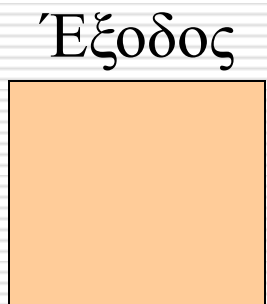
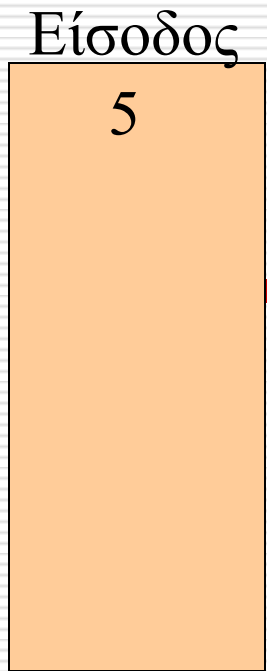




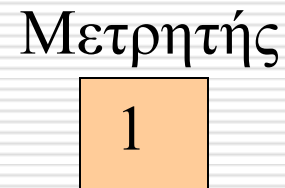
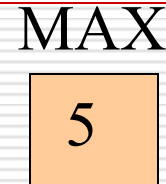
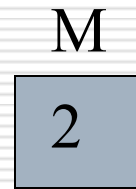
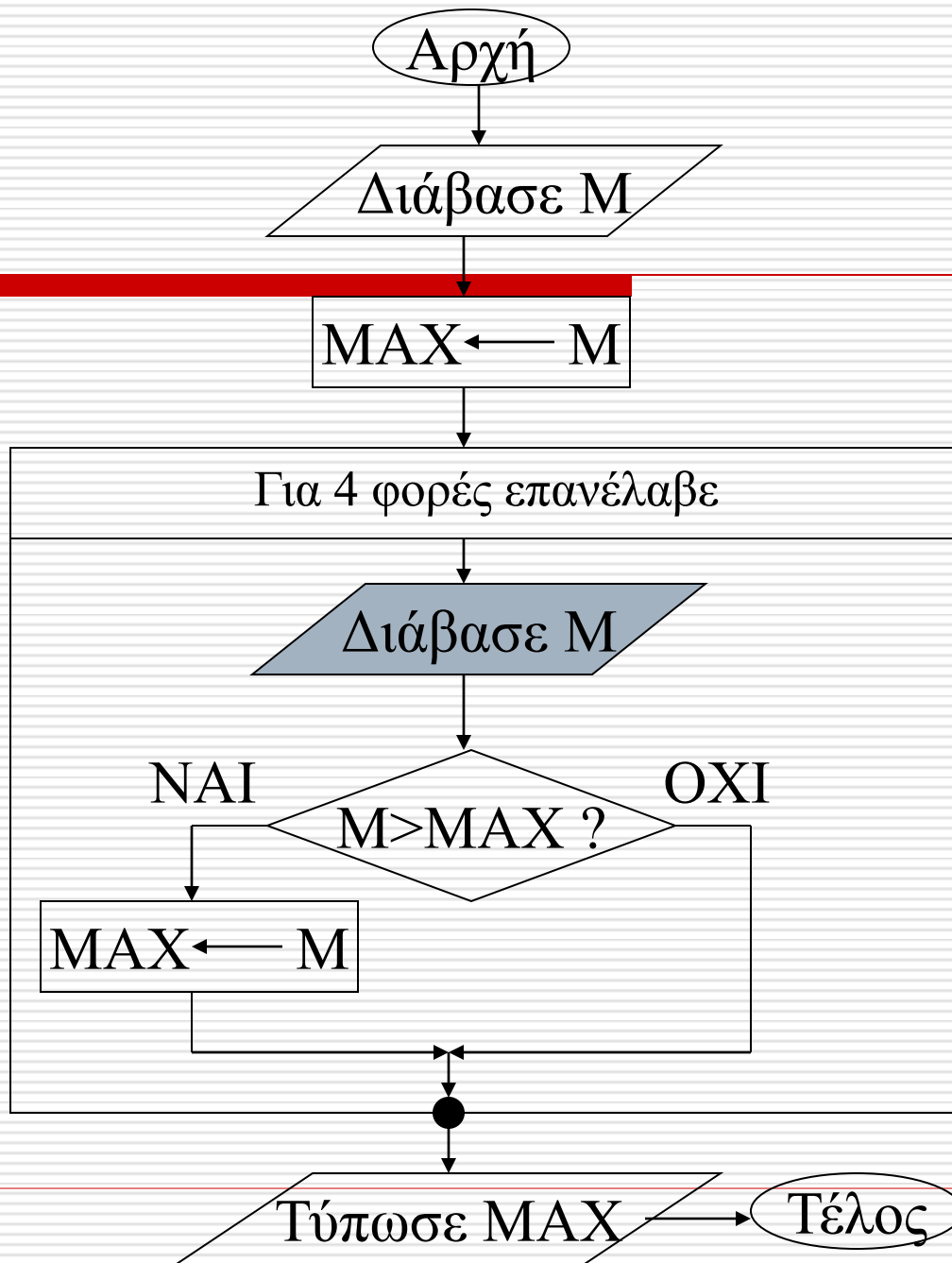
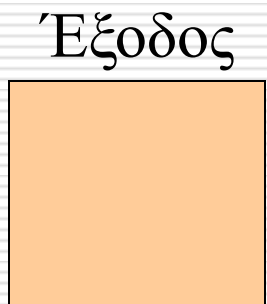
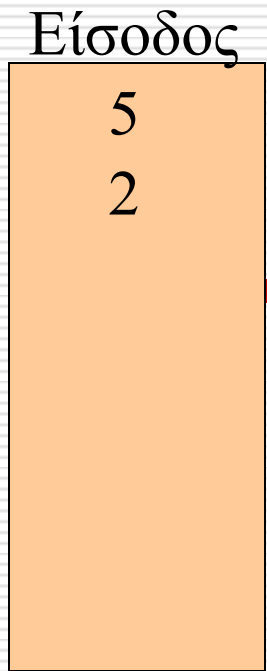


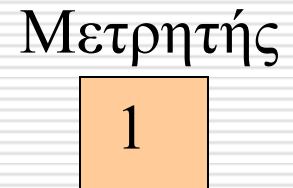
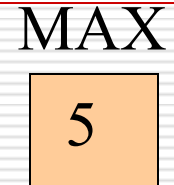
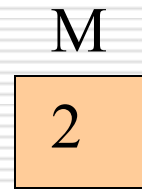
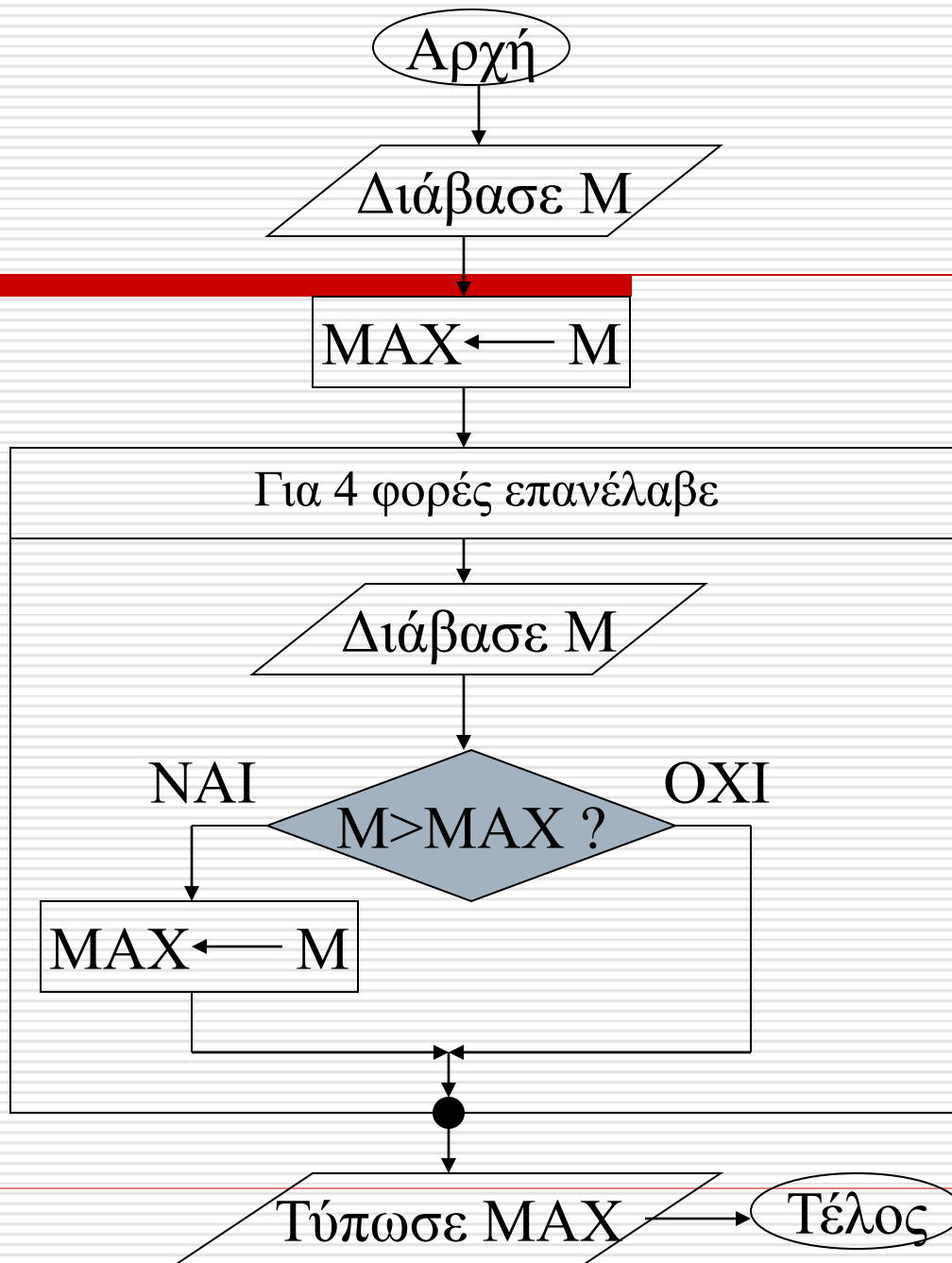
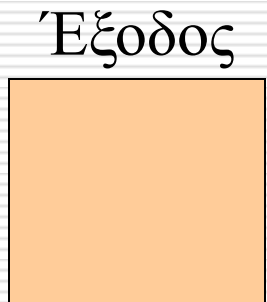
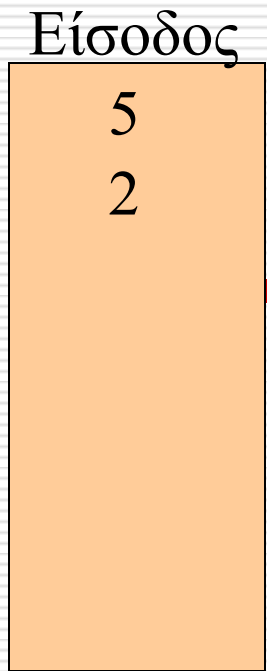


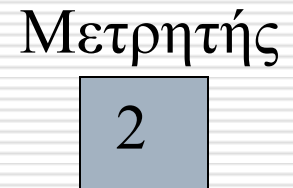
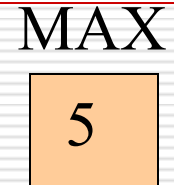
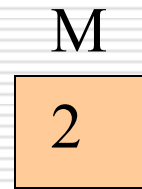
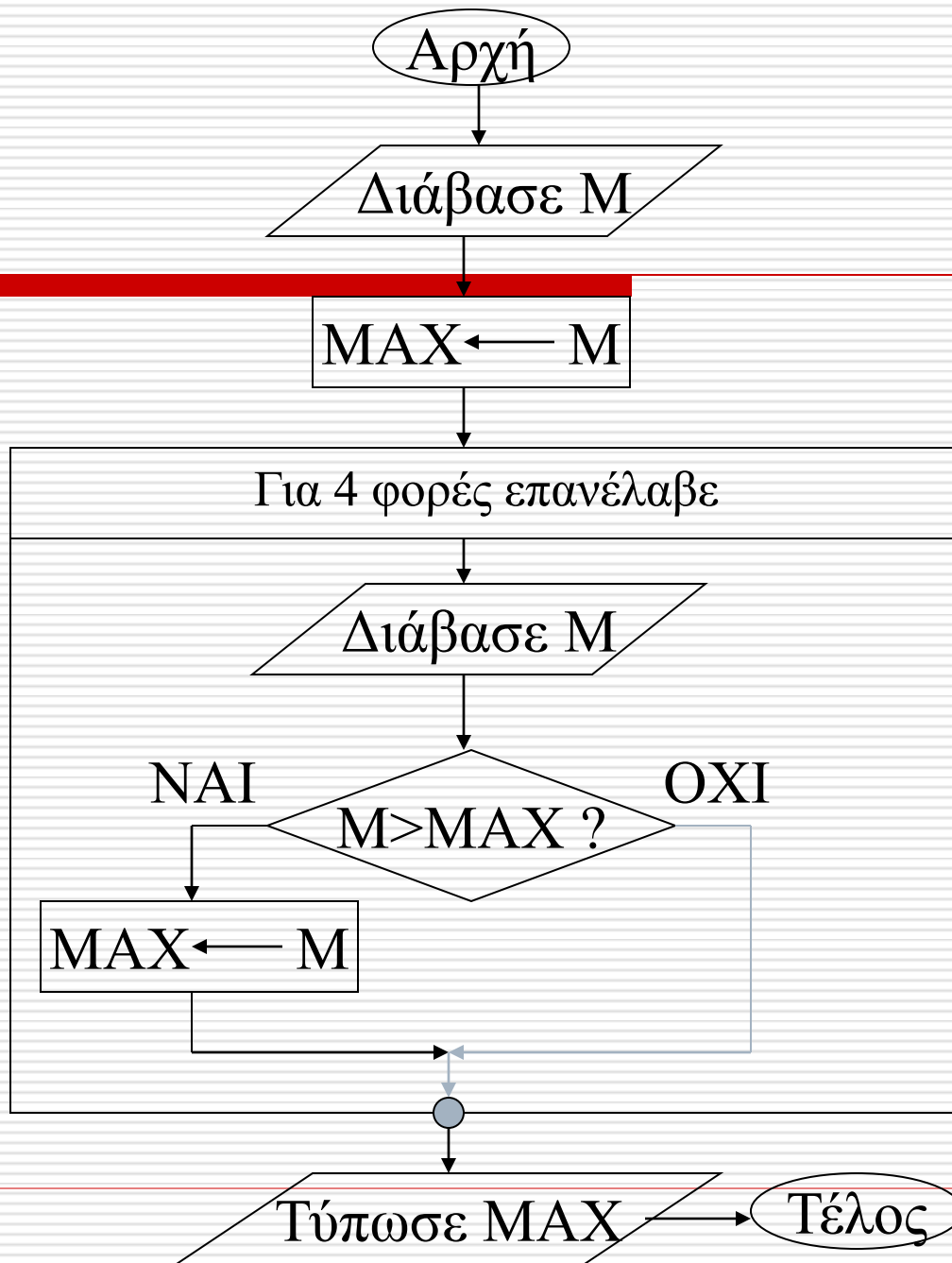
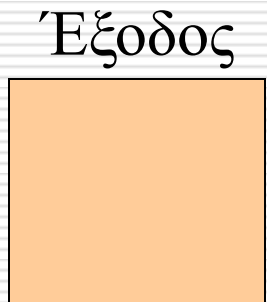
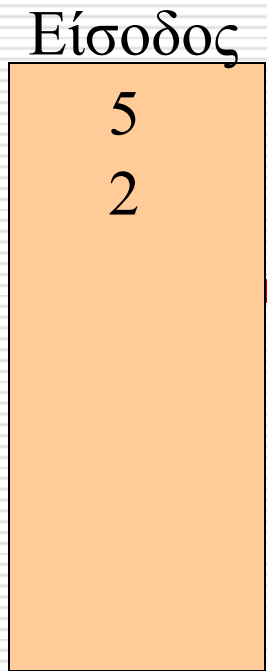




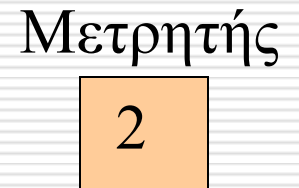
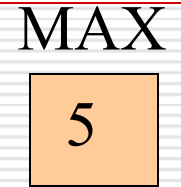
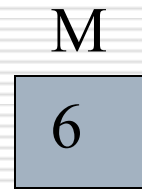
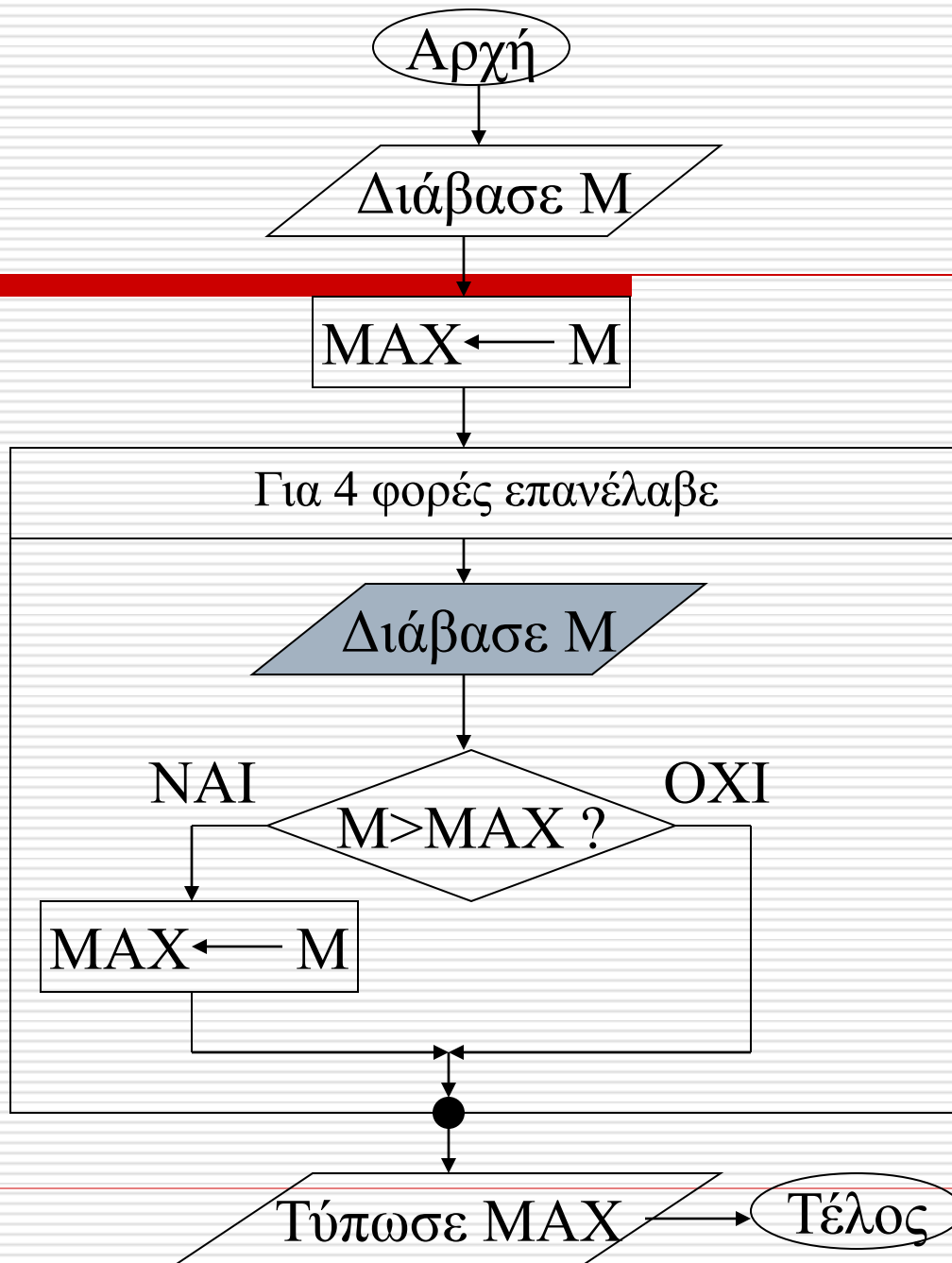
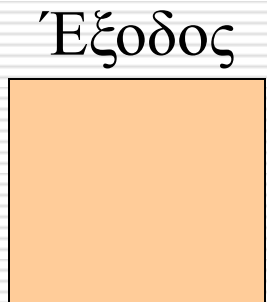
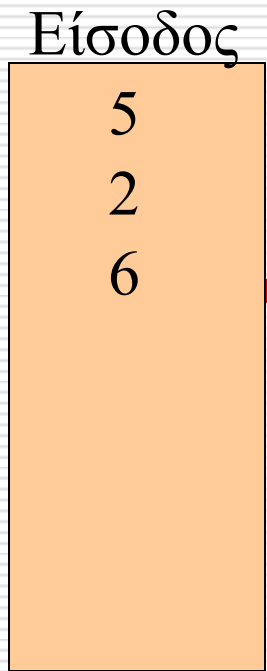
$1 \leq 4?$

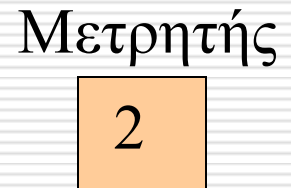
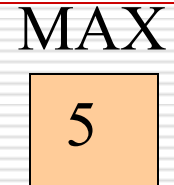
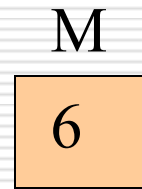
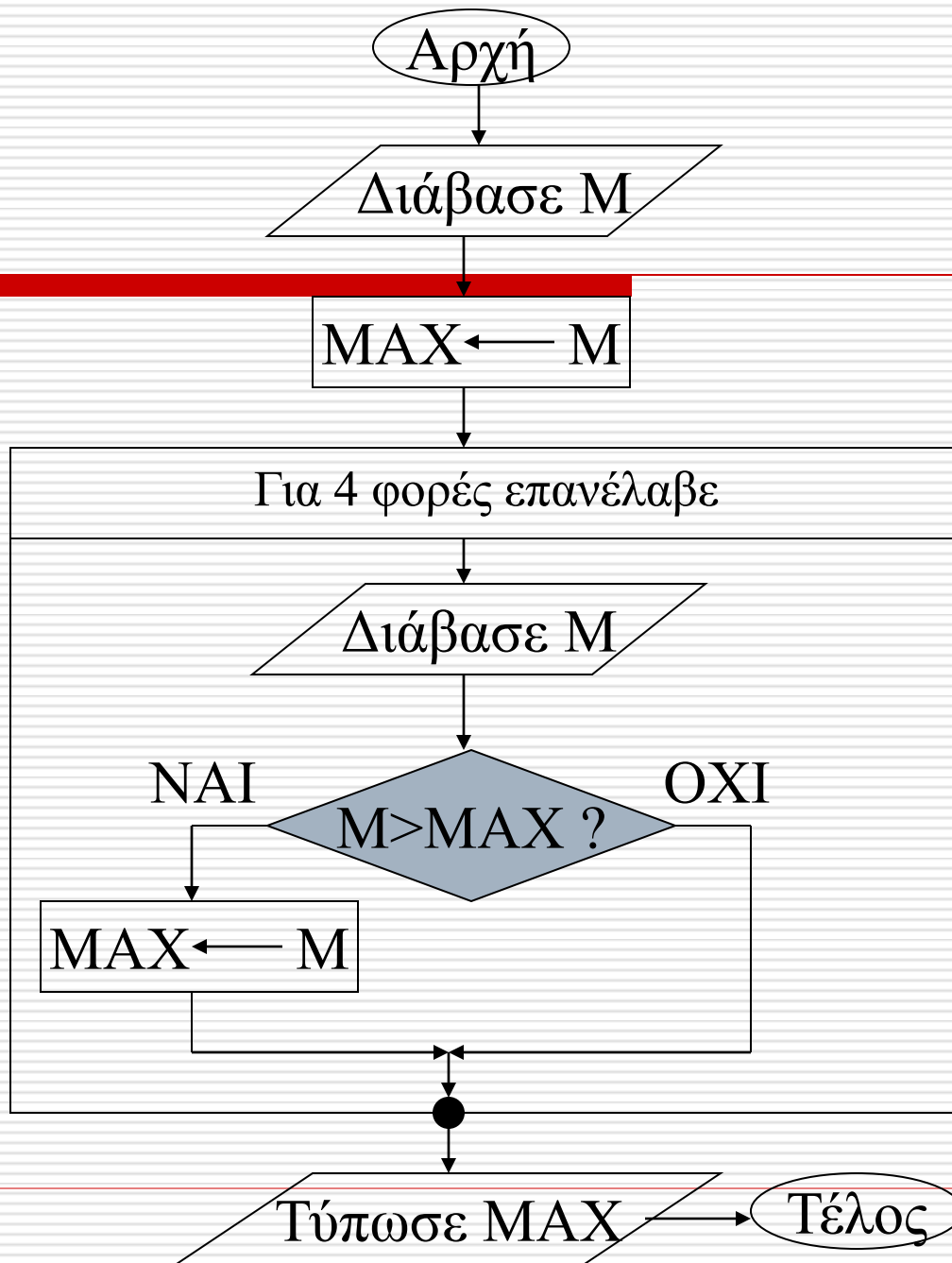
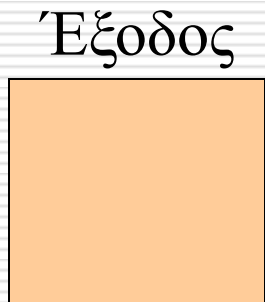
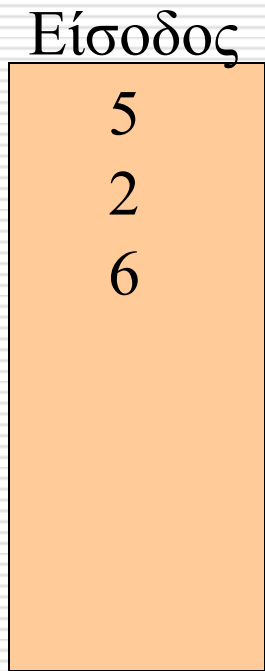


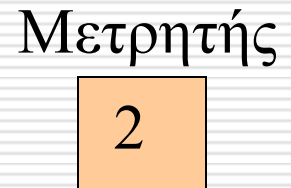
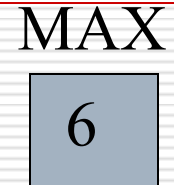
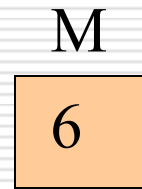
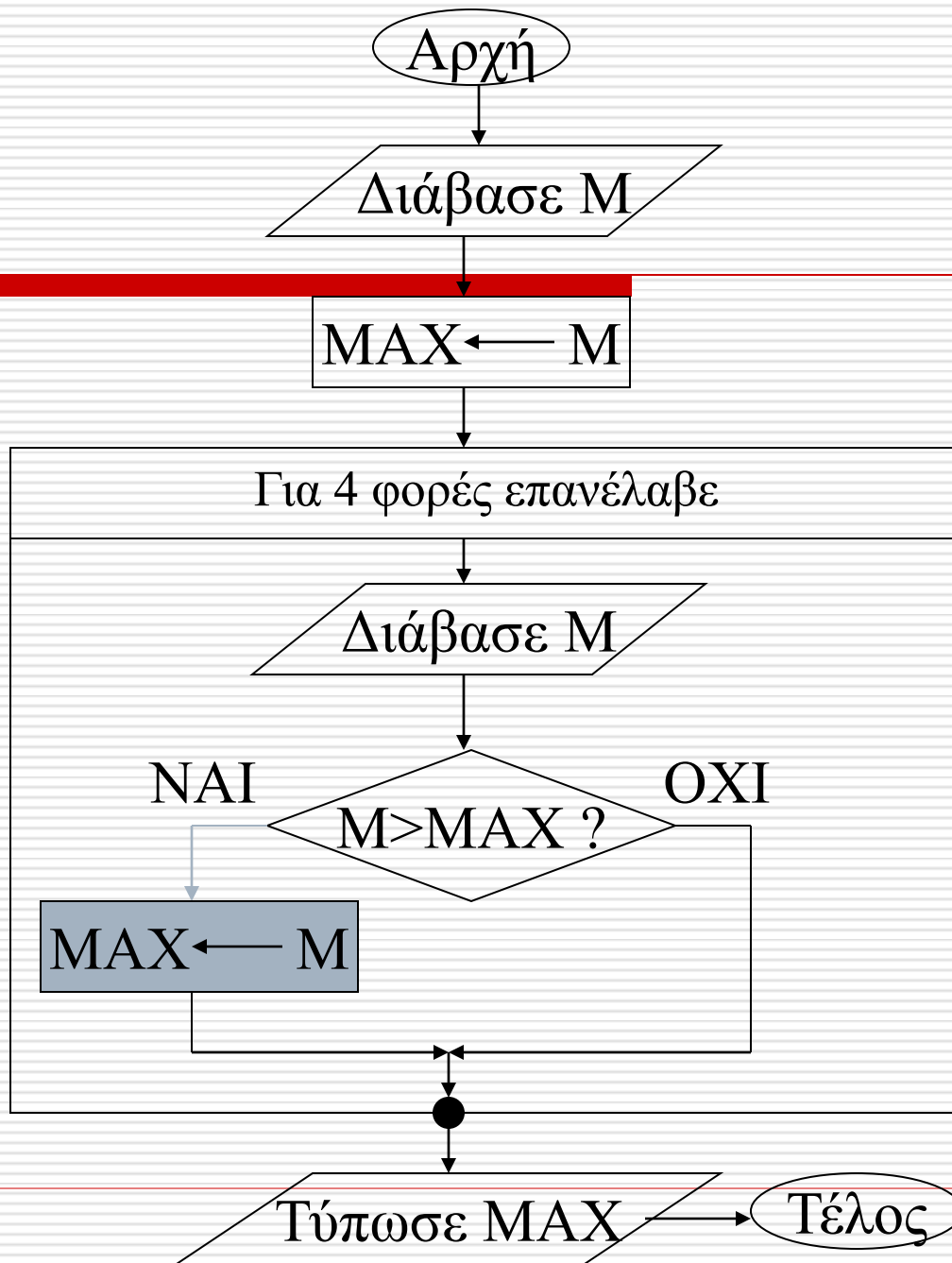
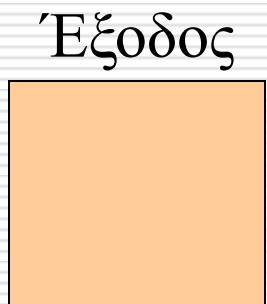
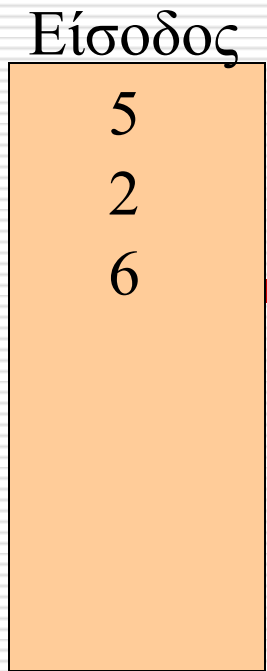


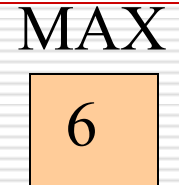
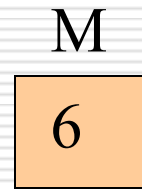
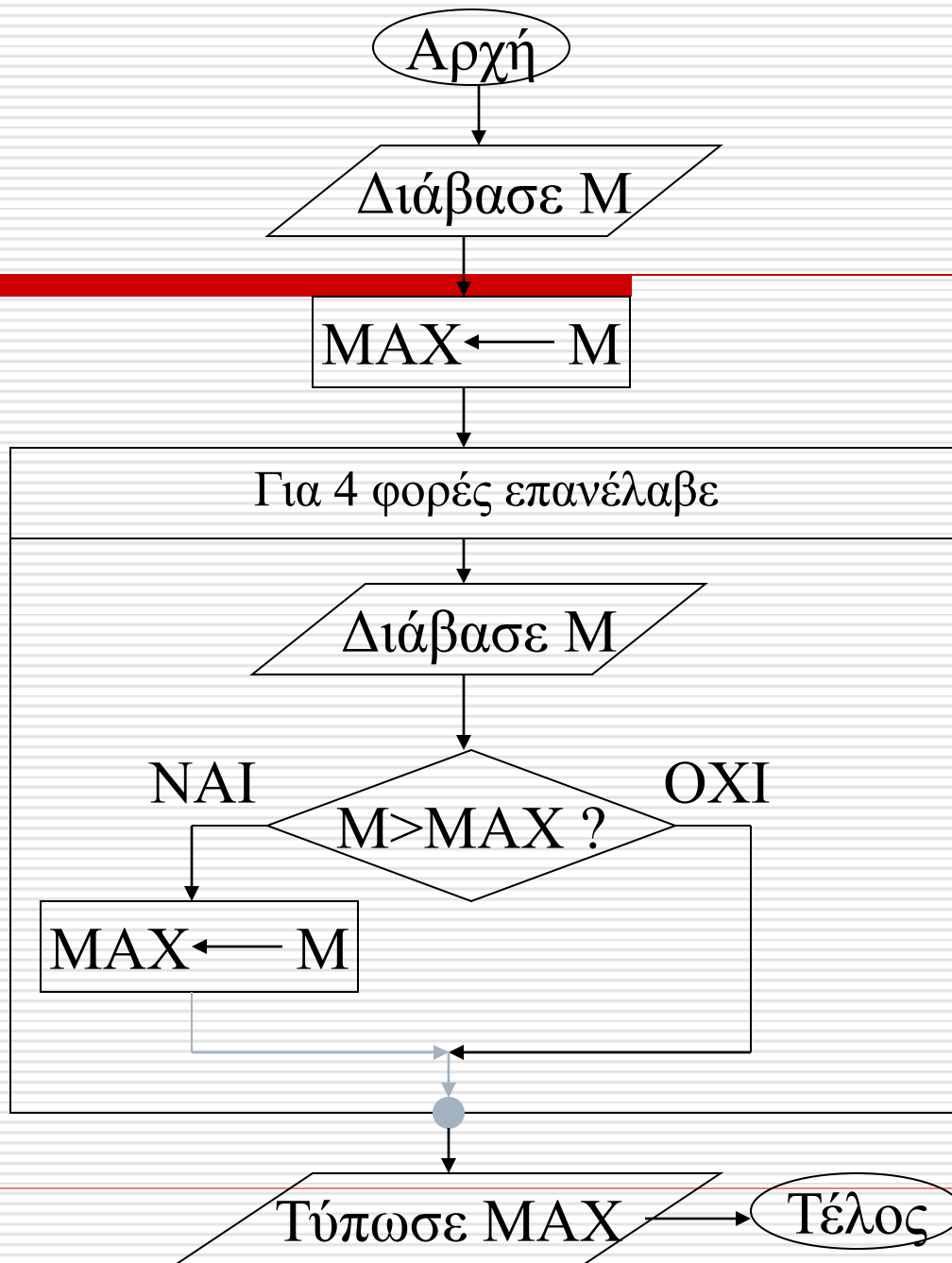
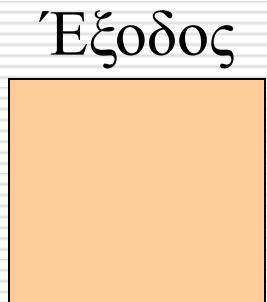
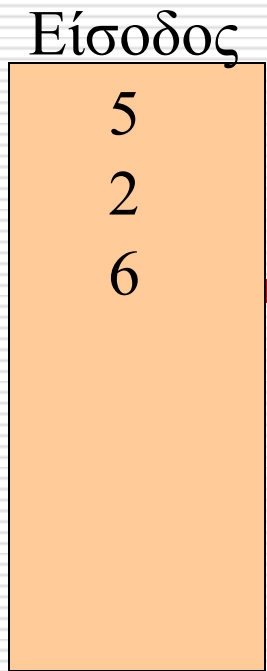


$2 \leq 4?$

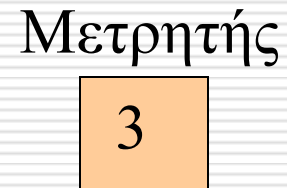
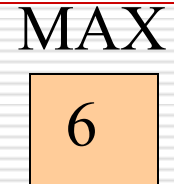
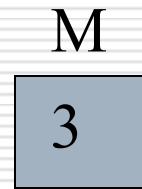
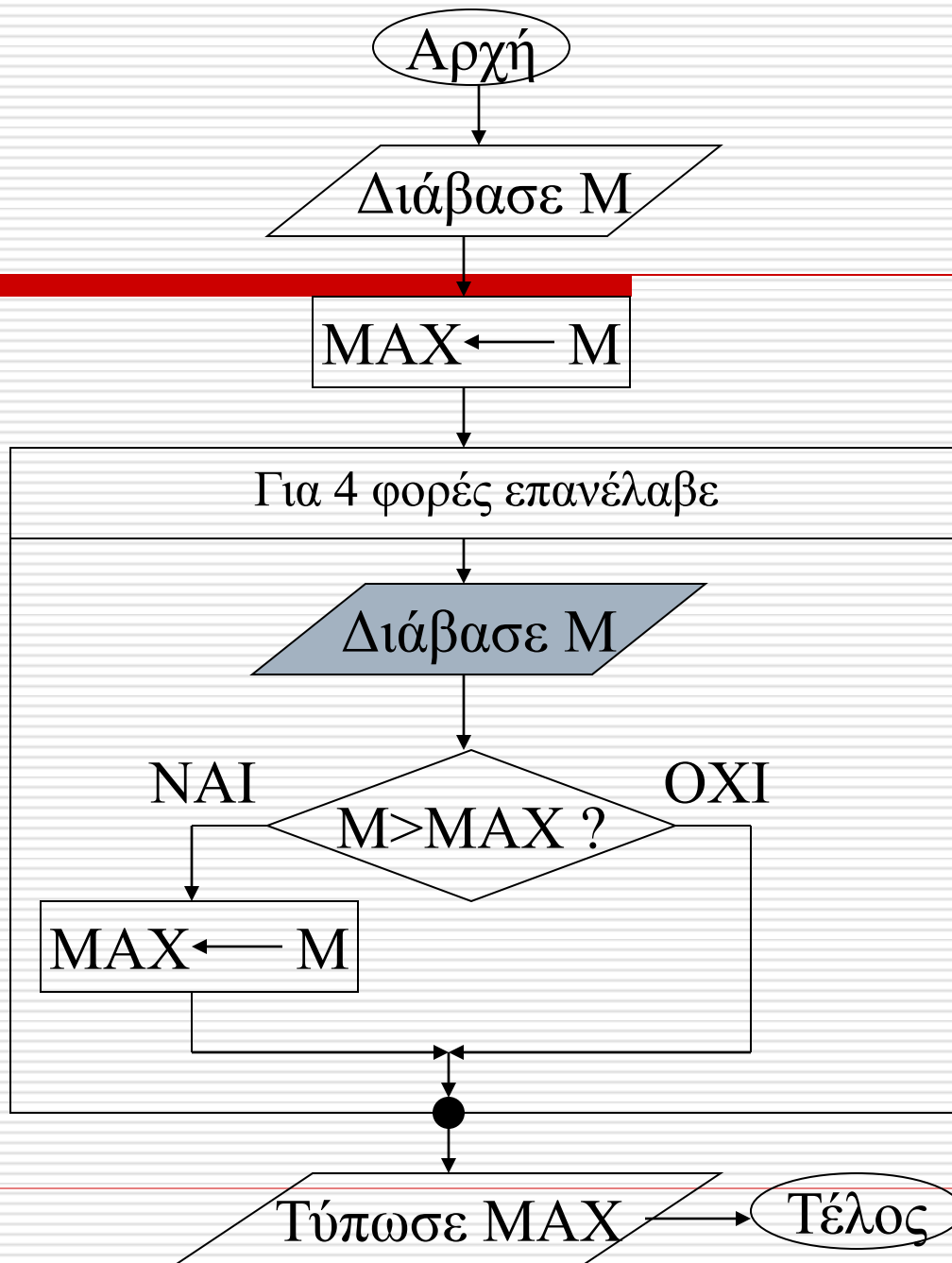
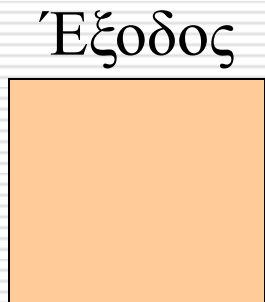
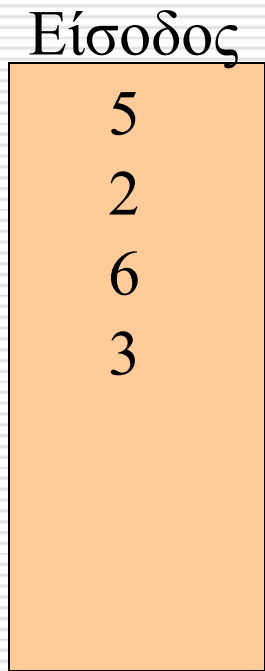


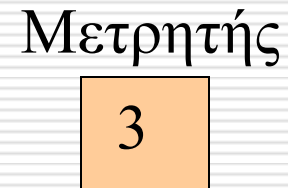
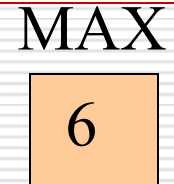
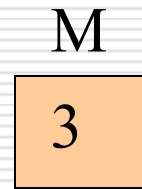
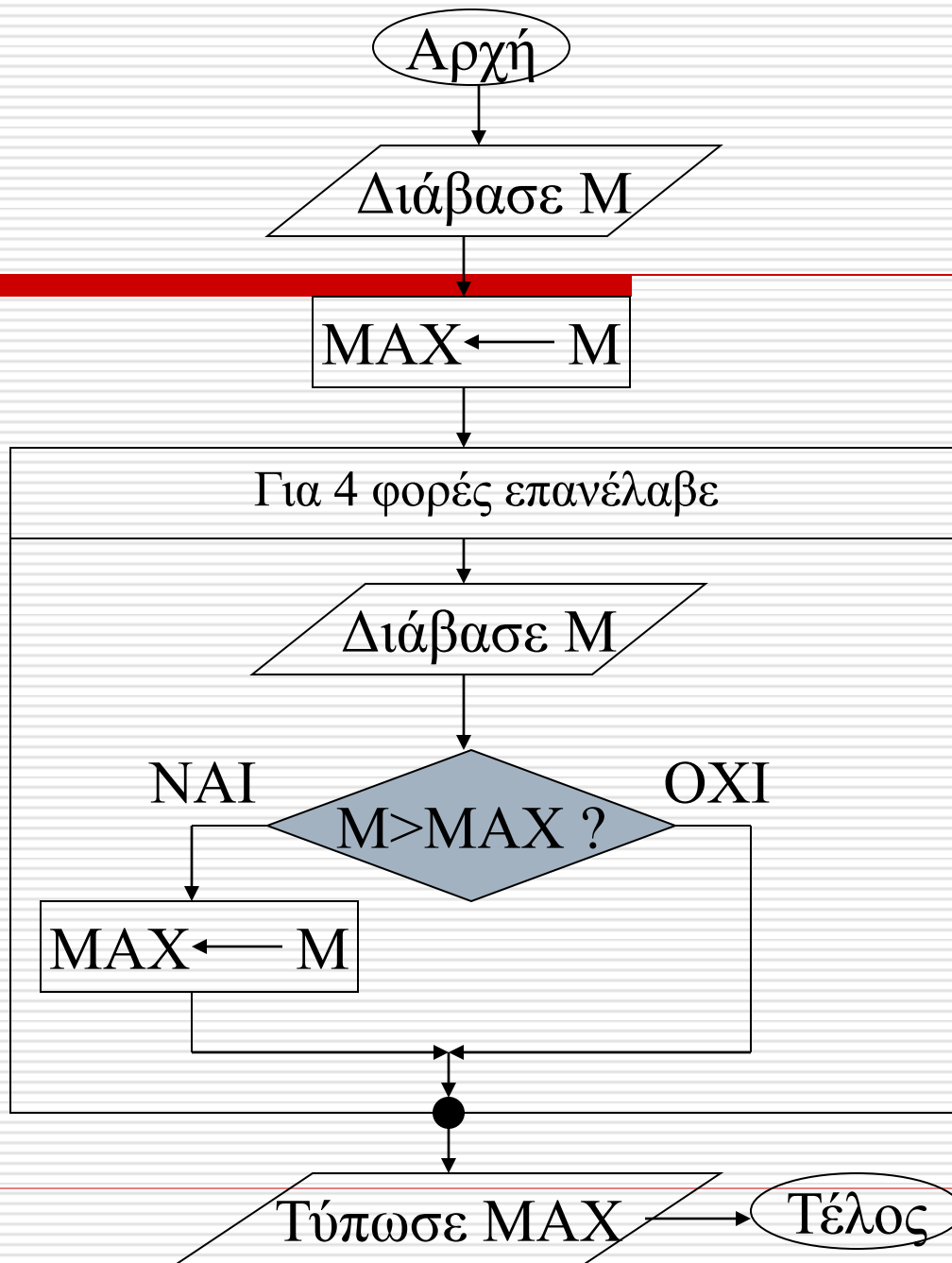
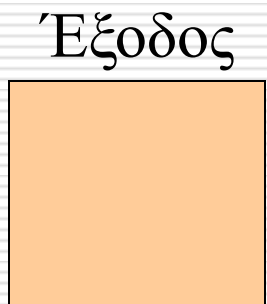
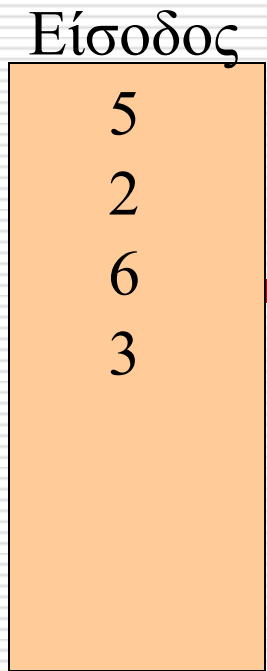


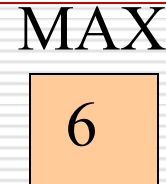
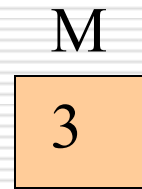
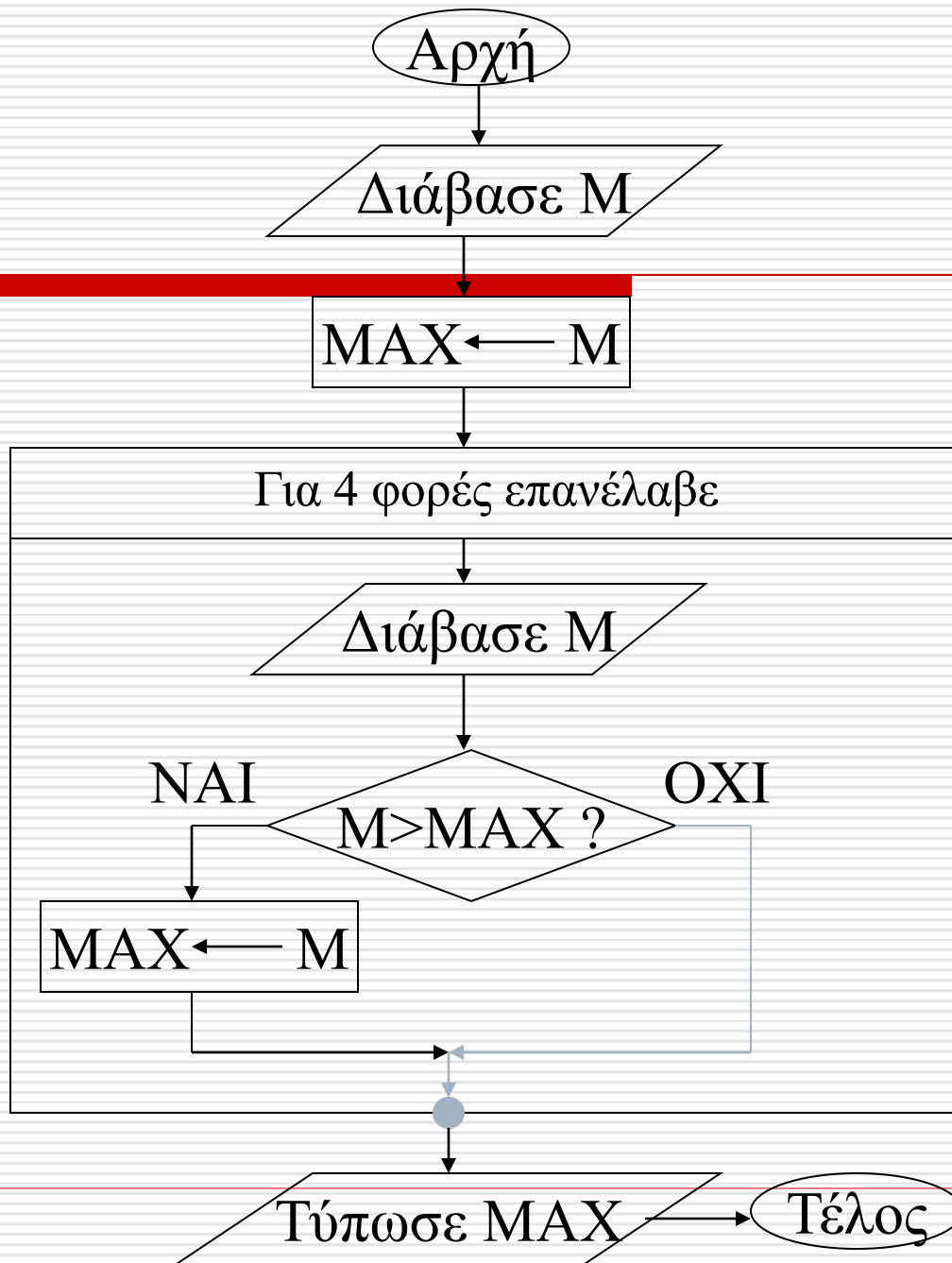
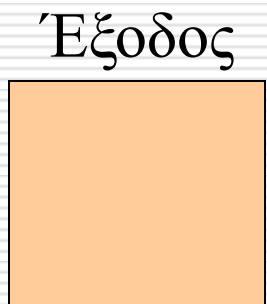
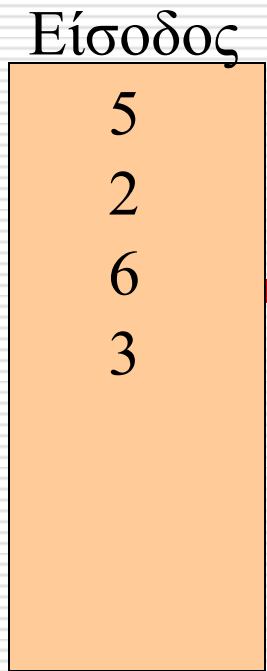




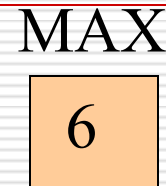
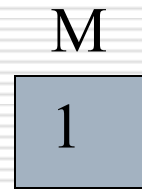
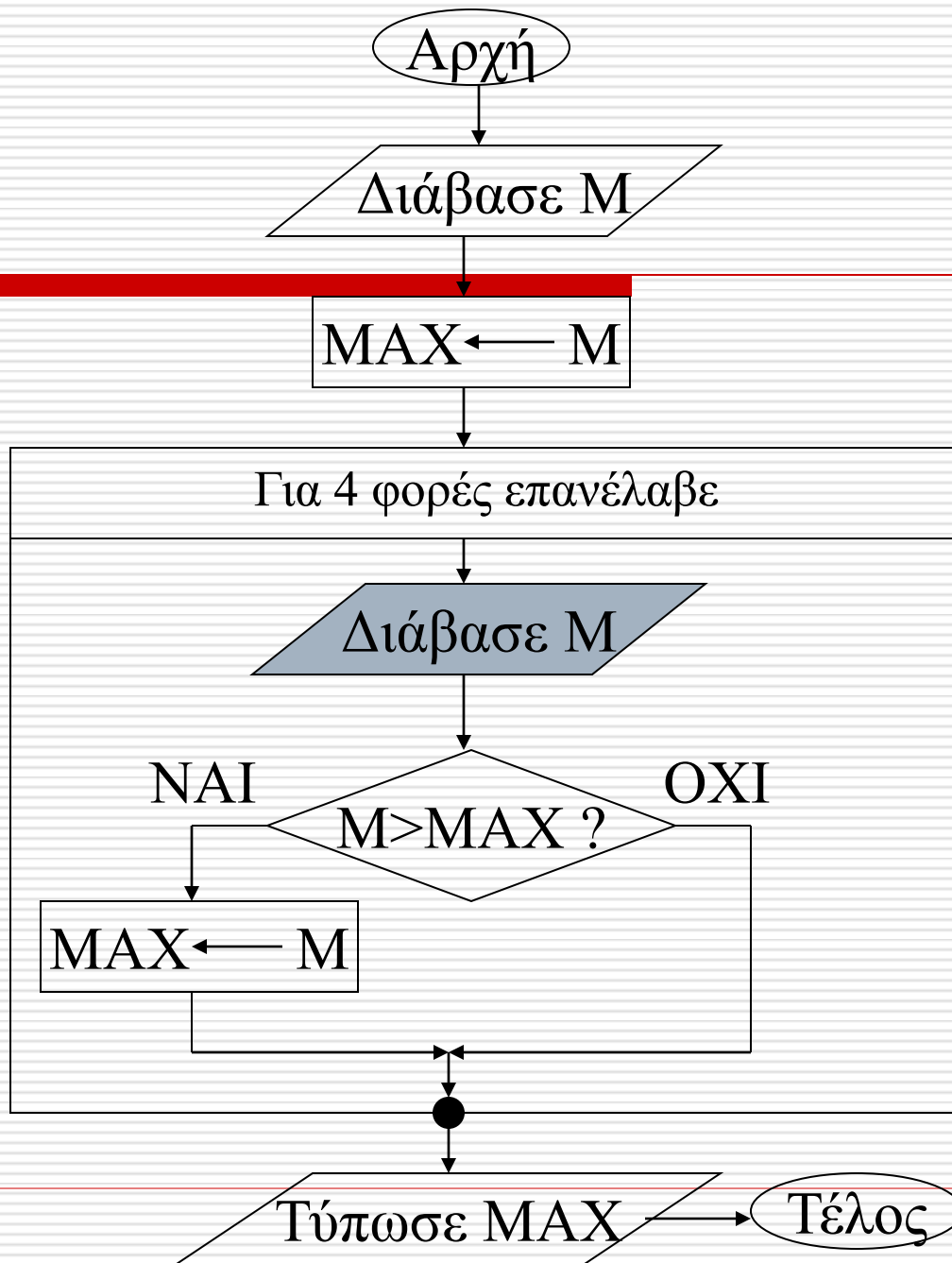
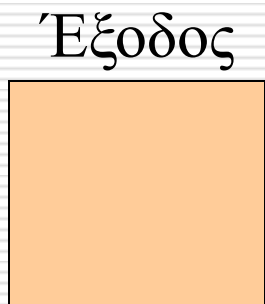
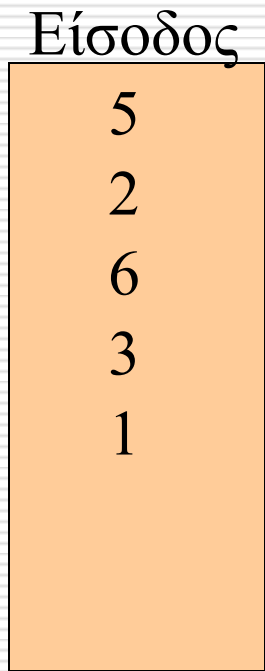
$3 \leq 4 ?$

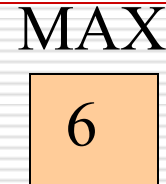
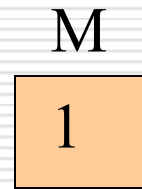
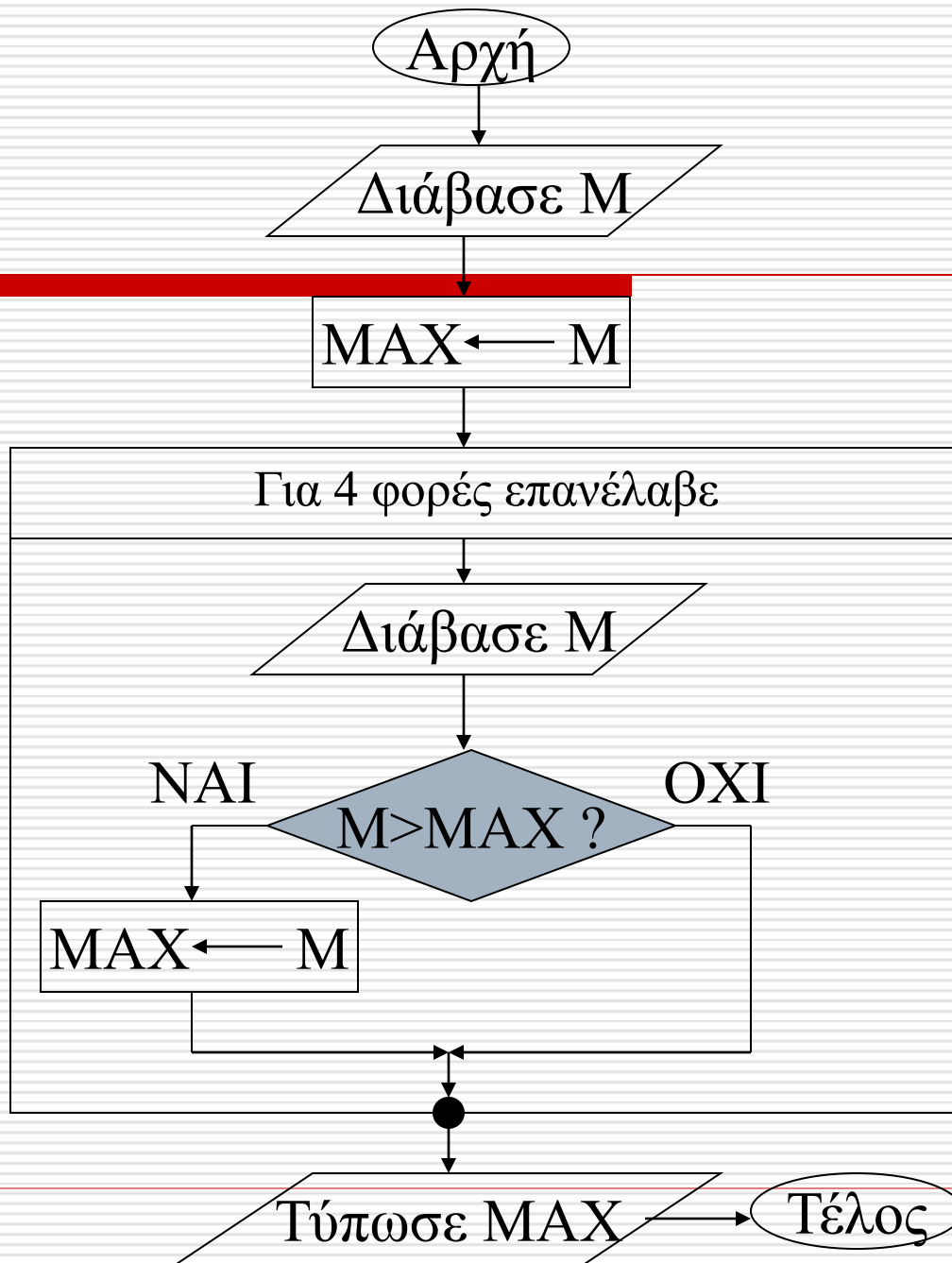
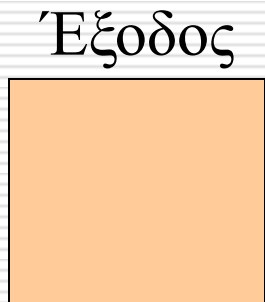
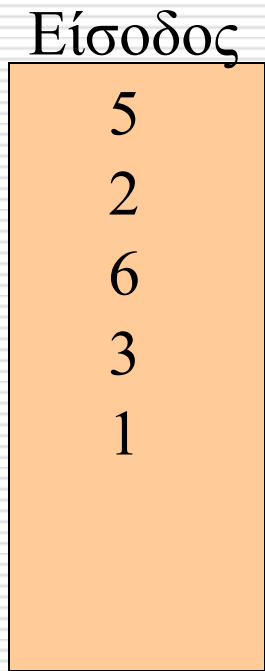


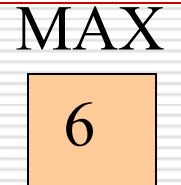
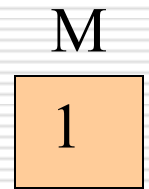
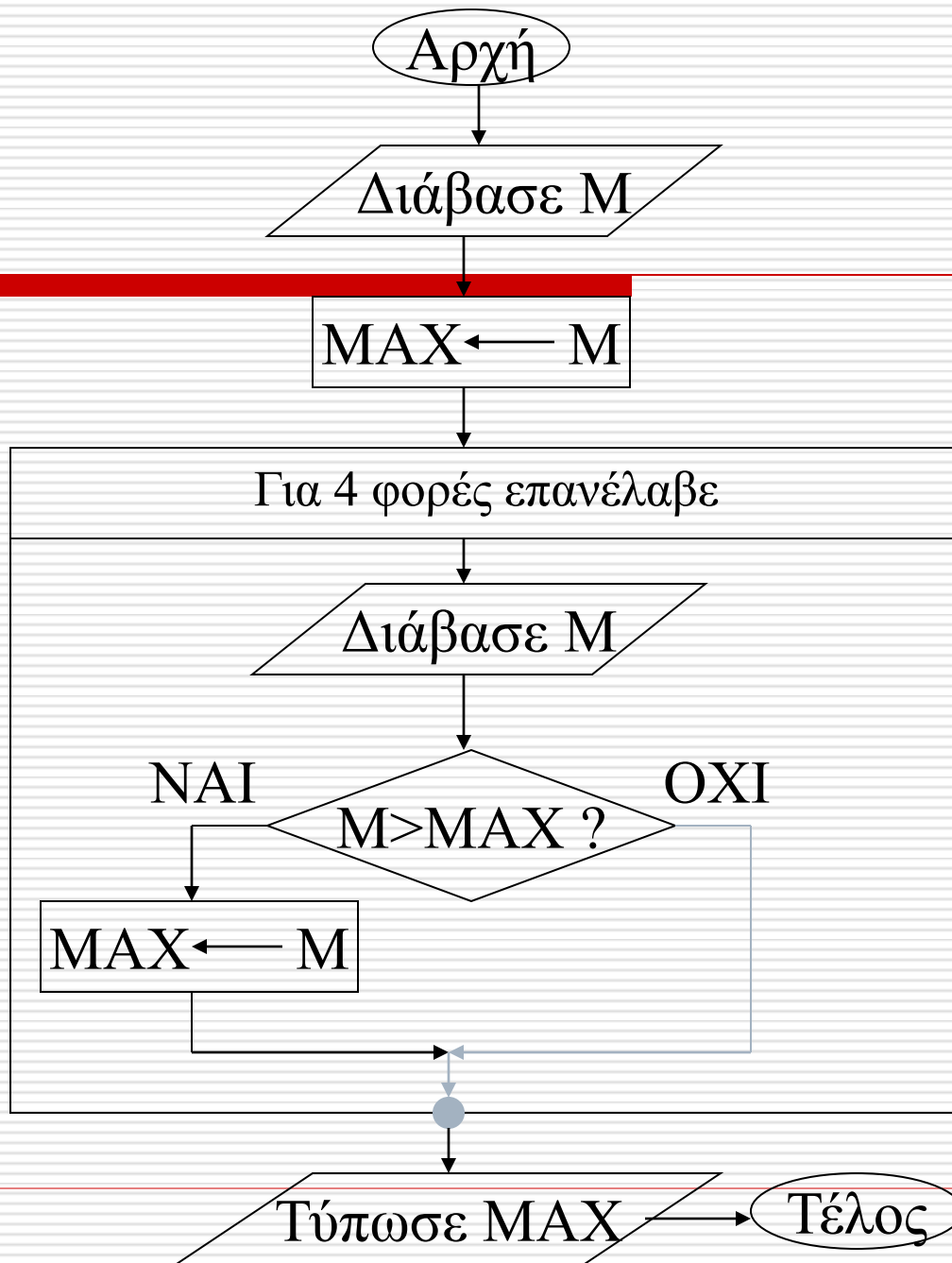
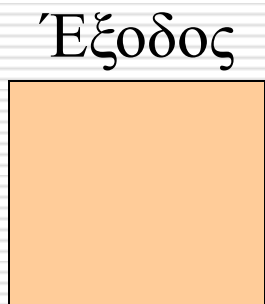
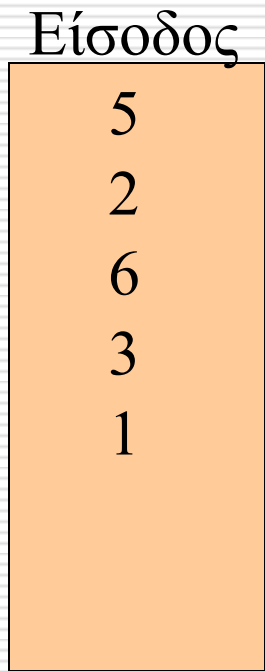




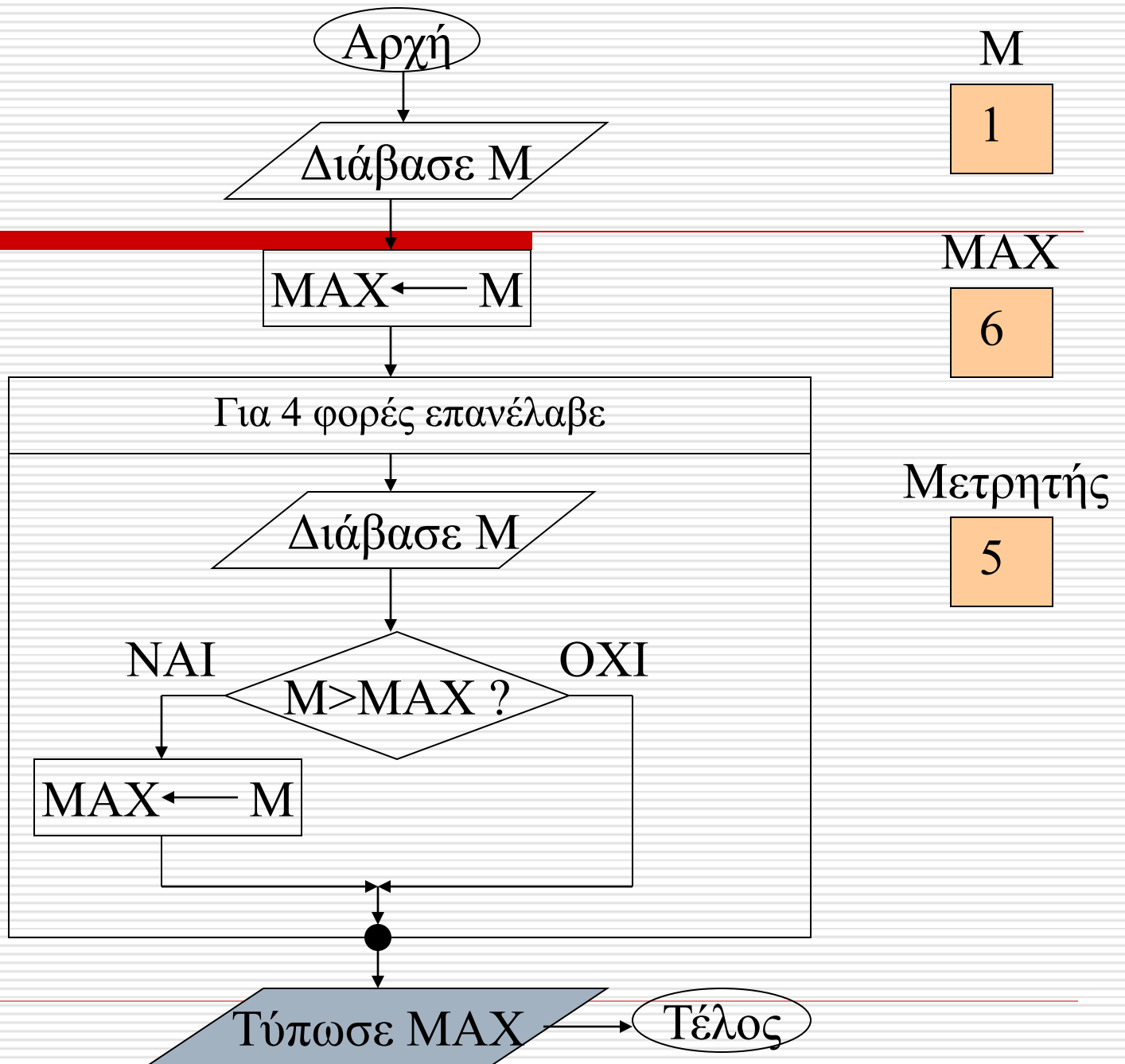
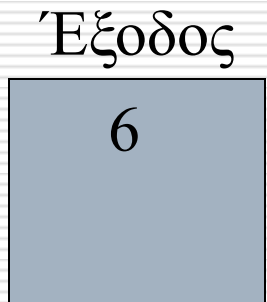
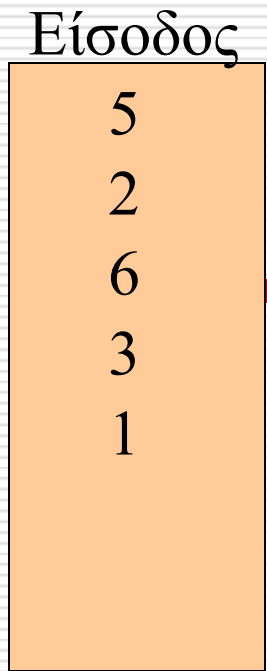
$4 \leq 4 ?$

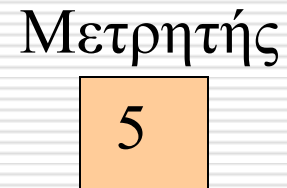
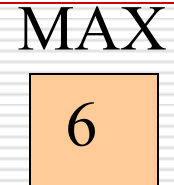
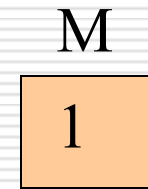
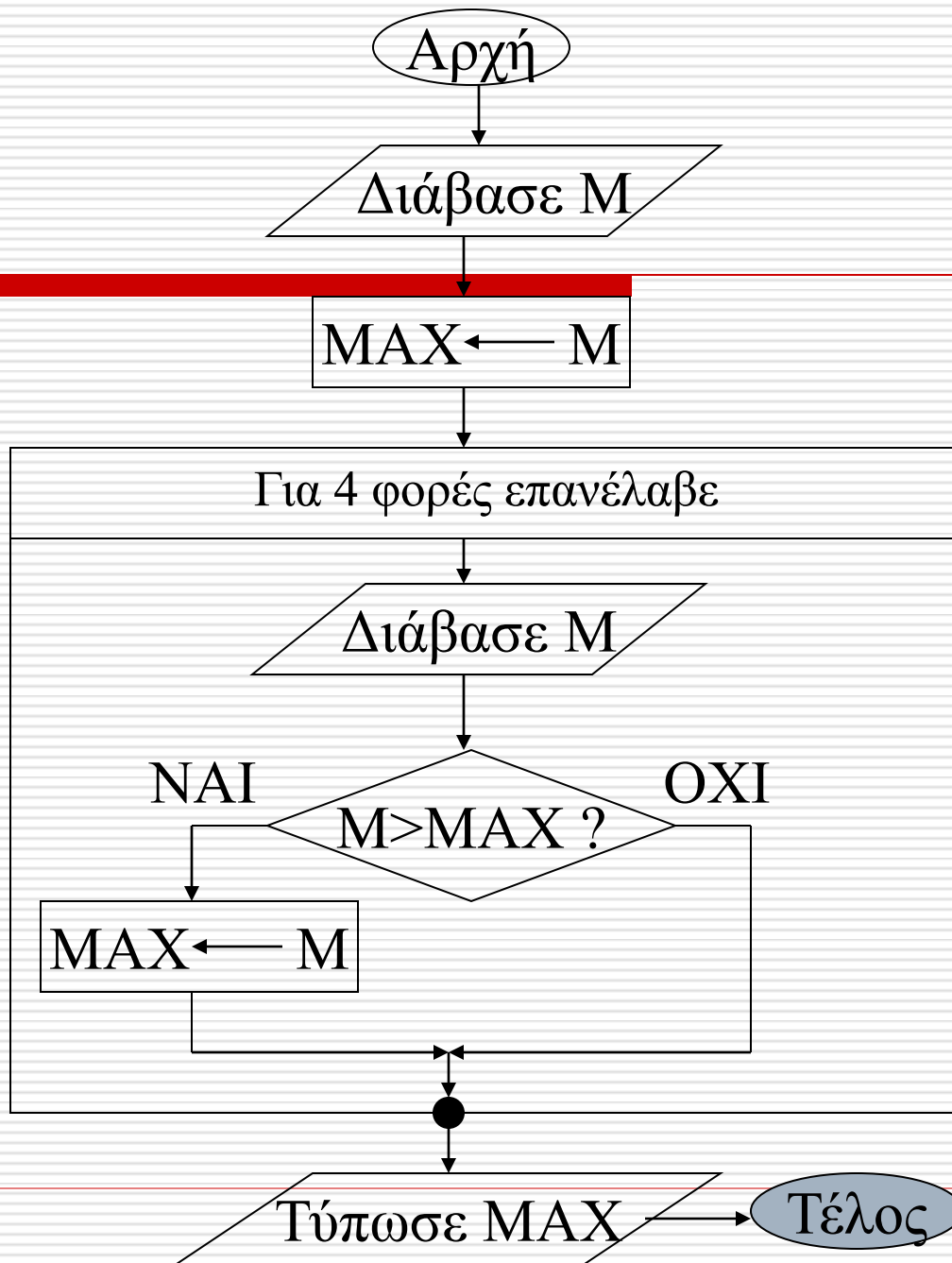
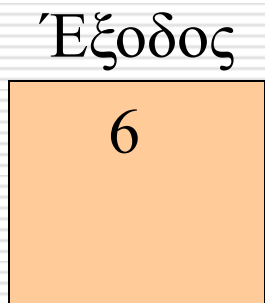
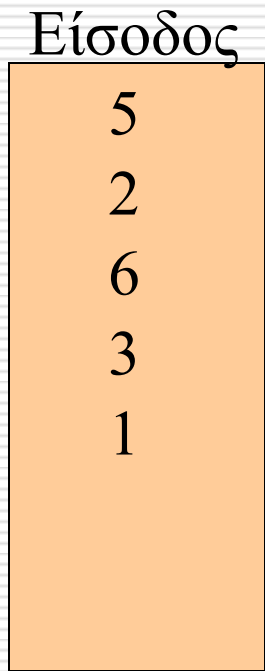






$5 \leq 4 ?$





Άσκηση -4

- Να γραφεί πρόγραμμα το οποίο
 - Διαβάζει 10 αριθμούς.
 - Υπολογίζει το άθροισμα των αριθμών αυτών.
 - Μετά το τέλος εισαγωγής των στοιχείων το πρόγραμμα εμφανίζει στην οθόνη
 - το άθροισμα των αριθμών αυτών
 - τον μεγαλύτερο από τους αριθμούς που δόθηκαν
 - τον μικρότερο από τους αριθμούς που δόθηκαν

Ασκηση-5

Να γραφεί πρόγραμμα που διαβάζει από την οθόνη τα στοιχεία παραγωγής για καθένα από **10** παραγωγούς:

- Κωδικός παραγωγού (ακέραιος)
- είδος παραγωγής (με τιμές 1=τομάτα, 2=πιπερια 3=μελιτζάνα).
- ποσότητα σε κιλά (ακέραιος).
- τιμή ανά κιλό

Στη συνέχεια υπολογίζει και εμφανίζει την αξία του προϊόντος

Στο τέλος της εισαγωγής των στοιχείων εμφανίζει οθόνη:

- τη συνολική ποσότητα παραγωγής για κάθε είδος
- τον παραγωγό με τη μεγαλύτερη αξία παραγωγής
- το προϊόν με τη μεγαλύτερη συνολική αξία (όλοι οι παραγωγοί)
- το ποσοστό παραγωγής τομάτας (ποσότητα) ως προς τη συνολική παραγωγή όλων των προϊόντων.

Άσκηση-6

- Για τον υπολογισμό της αξίας των προϊόντων που παραδίδουν οι παραγωγοί εσπεριδοειδών στον Συνεταιρισμό τους λαμβάνεται υπόψη ο παρακάτω πίνακας:

	Πορτοκάλια Τιμή/κιλό	Μανταρίνια Τιμή/κιλό
Ποιότητα -A	0.30€	0.45€
Ποιότητα -B	0.25€	0.30€

Άσκηση-6

- ❑ Να γραφεί πρόγραμμα σε **Python** που:
- ❑ Διαβάζει από την οθόνη τα στοιχεία παραγωγής **N** παραγωγών
 - Ονοματεπώνυμο παραγωγού (**eponymo**: χαρακτήρες)
 - Κωδικός προϊόντος (**eidoss**: ακέραιος με τιμές 1=πορτοκάλια 2= μανταρίνια).
 - Ποιότητα (**poiotita**: χαρακτήρας, με τιμές **A** ή **B**)
 - Ποσότητα σε κιλά (**posot**: ακέραιος)
- ❑ Υπολογίζει την αξία (**axia**) που δικαιούται ο παραγωγός
- ❑ Γράφει στην οθόνη το ονοματεπώνυμο του παραγωγού και την αξία που δικαιούται.
- ❑ Μετά το τέλος της διαδικασίας το σύστημα εμφανίζει στην οθόνη:
 - Το πλήθος (**plithosp**) των πορτοκαλοπαραγωγών.
 - Το συνολικό (**saxiap**) ποσό που δόθηκε στους πορτοκαλοπαραγωγούς.
 - Την συνολική ποσότητα παραγωγής (**sparagm**) σε μανταρίνια
 - Το ονοματεπώνυμο του παραγωγού με τη **μεγαλύτερη αξία παραγωγής σε μανταρίνια ποιότητας A**.