

**ΕΝΔΕΙΚΤΙΚΕΣ ΑΠΑΝΤΗΣΕΙΣ ΔΙΑΓΩΝΙΣΜΑΤΟΣ  
ΠΛΗΡΟΦΟΡΙΚΗΣ Γ' ΓΕΝΙΚΟΥ ΛΥΚΕΙΟΥ**

**ΕΠΙΜΕΛΕΙΑ ΔΙΑΓΩΝΙΣΜΑΤΟΣ: ΚΑΤΕΡΙΝΑ ΔΙΝΑΡΔΟΥ**

**ΘΕΜΑ Α**

**A1.**

1. Λάθος      2. Σωστό      3. Λάθος      4. Λάθος      5. Λάθος

**A2.**

1. a, b, c, f      2. a, d      3. a, c, d, f

**A3. α)** Οι διαφορές της λίστας και του πίνακα είναι οι ακόλουθες:

1. Ο πίνακας είναι δομή τυχαίας προσπέλασης (υπάρχει πρόσβαση σε οποιοδήποτε κελί), σε αντίθεση με τη λίστα που είναι μία δομή ακολουθιακής ή σειριακής προσπέλασης. Για να φθάσουμε, δηλαδή, σε έναν κόμβο μιας λίστας πρέπει να περάσουμε από όλους τους προηγούμενους ξεκινώντας από τον πρώτο.

2. Ο πίνακας έχει σταθερό μέγεθος, το οποίο δηλώνεται εξαρχής κατά την υλοποίηση. Αυτό γίνεται, γιατί ο πίνακας είναι στατική δομή δεδομένων σε αντίθεση με τη λίστα που είναι δυναμική δομή και το μέγεθός της μπορεί να μεταβάλλεται καθώς εισέρχονται νέοι κόμβοι στη λίστα ή διαγράφονται κάποιοι άλλοι.

3. Οι κόμβοι της λίστας αποθηκεύονται σε μη συνεχόμενες θέσεις μνήμης σε αντίθεση με τους πίνακες, όπου τα στοιχεία αποθηκεύονται σε συνεχόμενες θέσεις μνήμης.

**β)** Στα **πλεονεκτήματα** των λιστών έναντι των πινάκων συγκαταλέγονται τα εξής:

1. Το δυναμικό τους μέγεθος.
2. Η ευκολία εισαγωγής και διαγραφής από οποιοδήποτε μέρος της λίστας, καθώς και η μη αναγκαιότητα δήλωσης του μεγέθους τους.

Στα **μειονεκτήματα** των λιστών έναντι των πινάκων περιλαμβάνονται τα εξής:

1. Η τυχαία πρόσβαση στη λίστα δεν επιτρέπεται. Είναι αδύνατο να φτάσετε στον ν-οστό κόμβο μιας απλά συνδεδεμένης λίστας χωρίς πρώτα να περάσετε από όλους τους κόμβους διαδοχικά μέχρι τον συγκεκριμένο κόμβο ξεκινώντας από τον πρώτο κόμβο. Εναλλακτικά, στην περίπτωση της διπλά συνδεδεμένης λίστας μπορείτε να ξεκινήσετε και από τον τελευταίο κόμβο. Επομένως, δεν μπορούμε να πραγματοποιήσουμε με αποτελεσματικό τρόπο δυαδική αναζήτηση σε συνδεδεμένες λίστες.

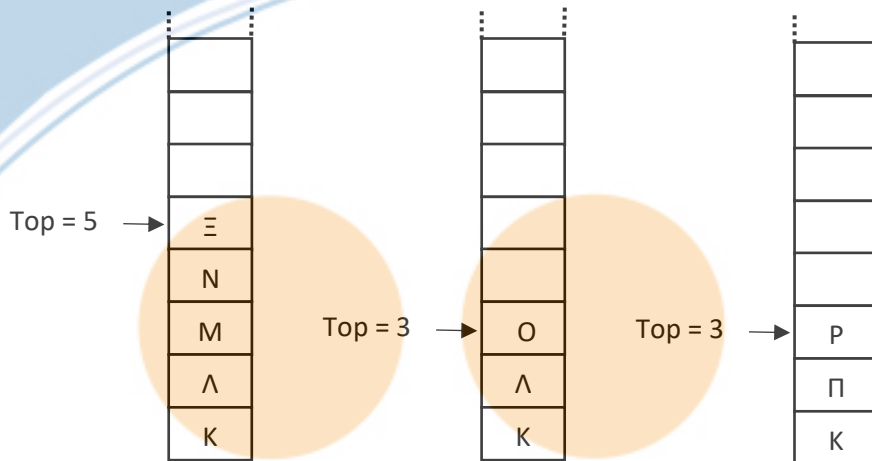
2. Οι συνδεδεμένες λίστες έχουν πολύ μεγαλύτερη επιβάρυνση από τους πίνακες, αφού οι συνδεδεμένοι κόμβοι της λίστας είναι δυναμικά κατανεμημένοι (οι οποίοι είναι λιγότερο αποτελεσματικοί στη χρήση της μνήμης) και κάθε κόμβος στη λίστα πρέπει, επιπλέον, να αποθηκεύει έναν πρόσθετο δείκτη που θα δείχνει στον επόμενο

κόμβο. Στην περίπτωση των διπλά συνδεδεμένων λιστών χρειαζόμαστε επιπλέον έναν δεύτερο δείκτη που θα δείχνει στον προηγούμενο κόμβο.

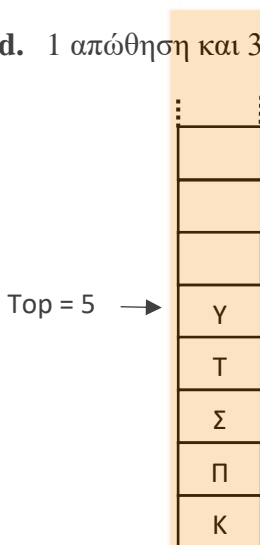
### ΘΕΜΑ Β

**B1.** Οι λειτουργίες που υλοποιούνται στη στοίβα είναι: ώθηση (κορυφή  $\leftarrow$  κορυφή + 1 ) και η απόθηση (κορυφή  $\leftarrow$  κορυφή - 1)

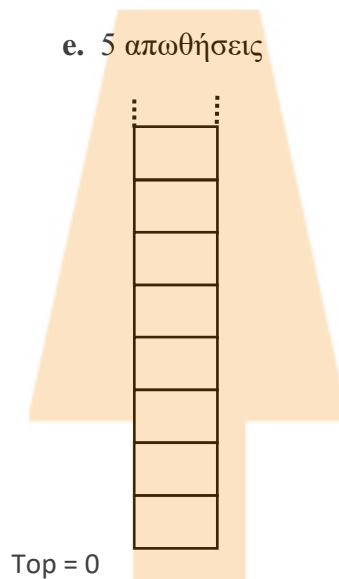
- a. Ωθηση 5 γραμμάτων    b. 3 αποθήσεις και 1 ώθηση    c. 2 αποθήσεις και 2 ωθήσεις



- d. 1 απόθηση και 3 ωθήσεις



- e. 5 αποθήσεις



Η τελική μορφή της στοίβας είναι μία άδεια στοίβα, αφού ο δείκτης κορυφή έχει τιμή μηδέν (top = 0).

### B2.

- a. Για να δημιουργηθεί η λέξη ΤΟΛΜΗ, θα πρέπει να γίνει διαγραφή των τεσσάρων αρχικών κόμβων στην λίστα και να τοποθετηθεί το γράμμα Τ ως πρώτος κόμβος. Στη συνέχεια η δεικτοδότηση θα παραμείνει ίδια μέχρι το κόμβο με το γράμμα Μ, όπου θα προστεθεί ως τελευταίος κόμβος αυτός με το γράμμα Η. Η δεικτοδότηση θα είναι η ακόλουθη λαμβάνοντας υπόψη ότι στον τελευταίο κόμβο με το γράμμα Η, θα

πρέπει το δεύτερο πεδίο να έχει την τιμή 0, σηματοδοτώντας το τέλος της λίστας:

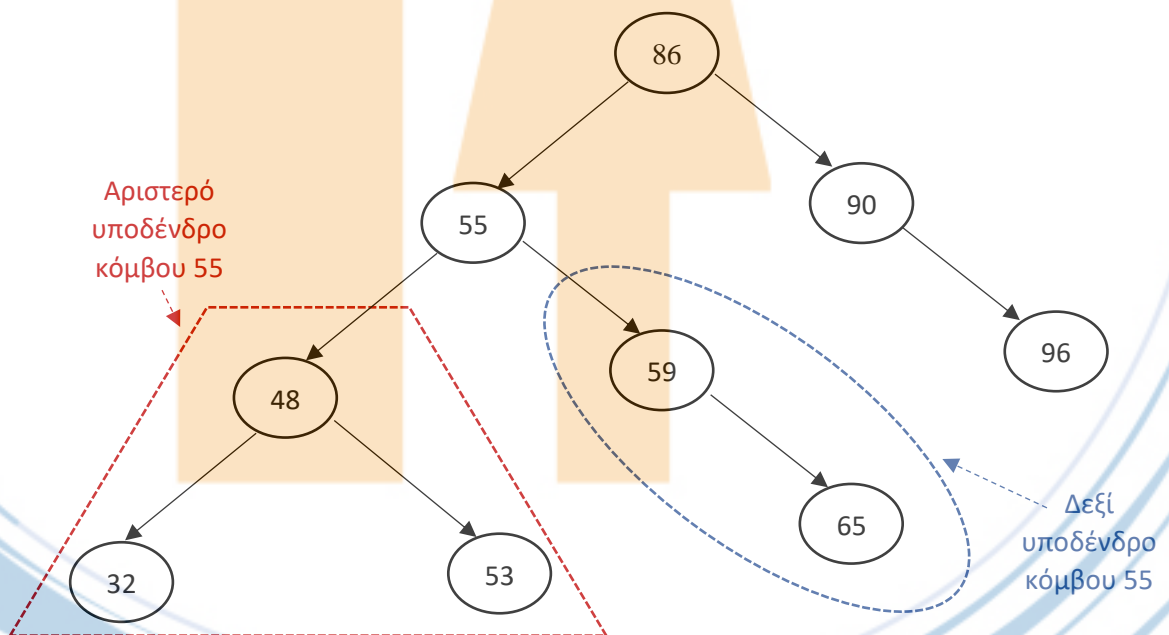
500	501	...	510	511	...	623	624	...	652	653	...	730	731	...
A		...	O	623	...	Λ	652	...	M	1067	...	E		...
850	851	...	910	911	...	924	925	...	945	946	...	1067	1068	...
P		...	I		...	T	510	...	M		...	H	0	...

b. Ομοίως για να δημιουργηθεί η λέξη ΛΗΜΜΑ, η δεικτοδότηση θα είναι η ακόλουθη:

500	501	...	510	511	...	623	624	...	652	653	...	730	731	...
A	0	...	O		...	Λ	1067	...	M	500	...	E		...
850	851	...	910	911	...	924	925	...	945	946	...	1067	1068	...
P		...	I		...	T		...	M	652	...	H	945	...

### B3.

- Το δένδρο είναι δυαδικό, γιατί κάθε κόμβος έχει το πολύ δύο παιδιά.
- Το δένδρο είναι διατεταγμένο, καθώς τα αδέρφια είναι ταξινομημένα σε αύξουσα διάταξη.
- Το αριστερό υποδένδρο του κόμβου 55 περιλαμβάνει τους κόμβους: 48, 32 και 53 (κόκκινο περίγραμμα). Το δεξί υποδένδρο του κόμβου 55 περιλαμβάνει τους κόμβους: 59 και 65 (μπλε περίγραμμα).



d. Είναι δυαδικό δένδρο αναζήτησης, γιατί καλύπτει τις προδιαγραφές ενός δυαδικού δένδρου αναζήτησης: το αριστερό υποδένδρο κάθε κόμβου περιέχει κόμβους με τιμή μικρότερη από αυτόν. Αντίστοιχα, το δεξί υποδένδρο κάθε κόμβου περιέχει κόμβους με τιμή μεγαλύτερη ή ίση με αυτόν.

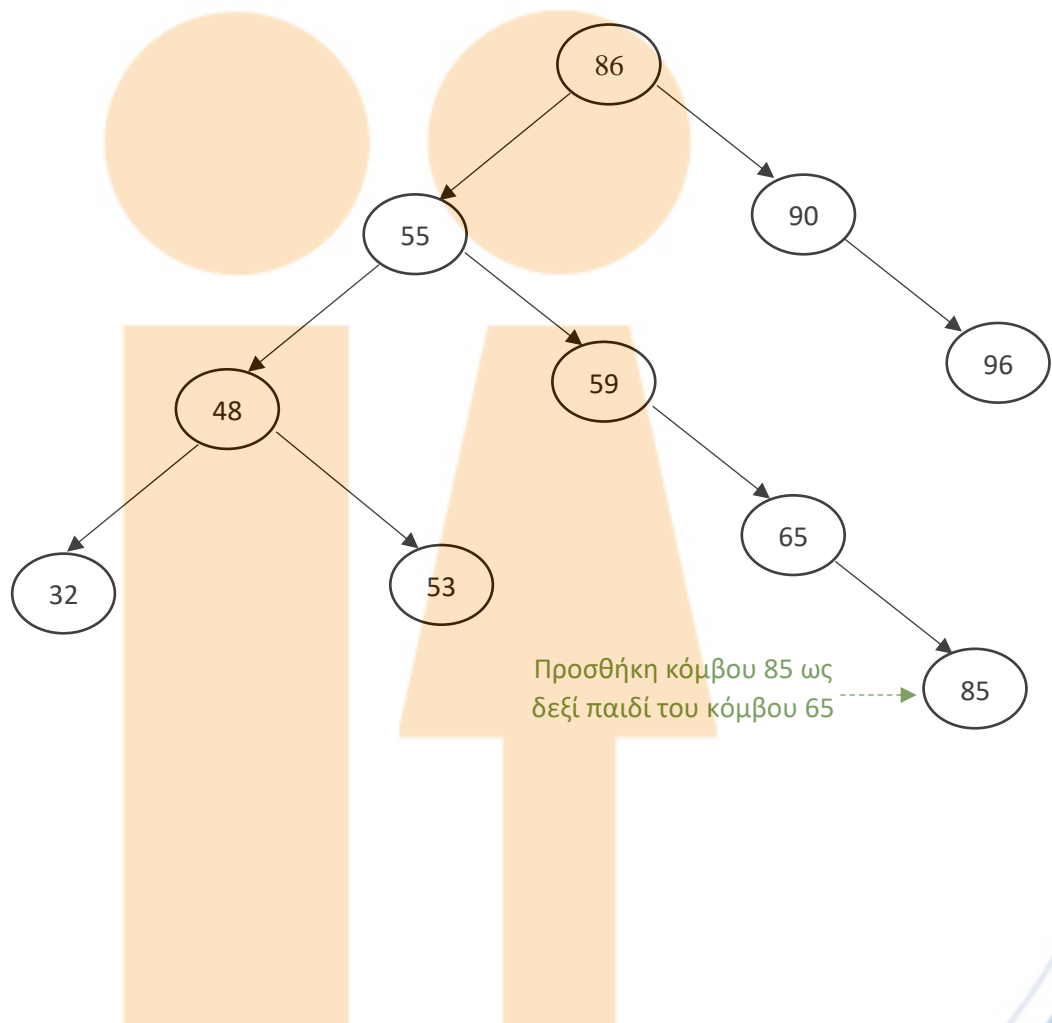
e. Ο αριθμός 82 αρχικά θα συγκριθεί με τη ρίζα (86). Αφού έχει τιμή μικρότερη, θα εστιάσουμε στο αριστερό υποδένδρο του 86.

Στη συνέχεια ο αριθμός 82 θα συγκριθεί με τον αριθμό 55. Αφού έχει τιμή μεγαλύτερη, θα εστιάσουμε στο δεξί υποδένδρο του κόμβου 55.

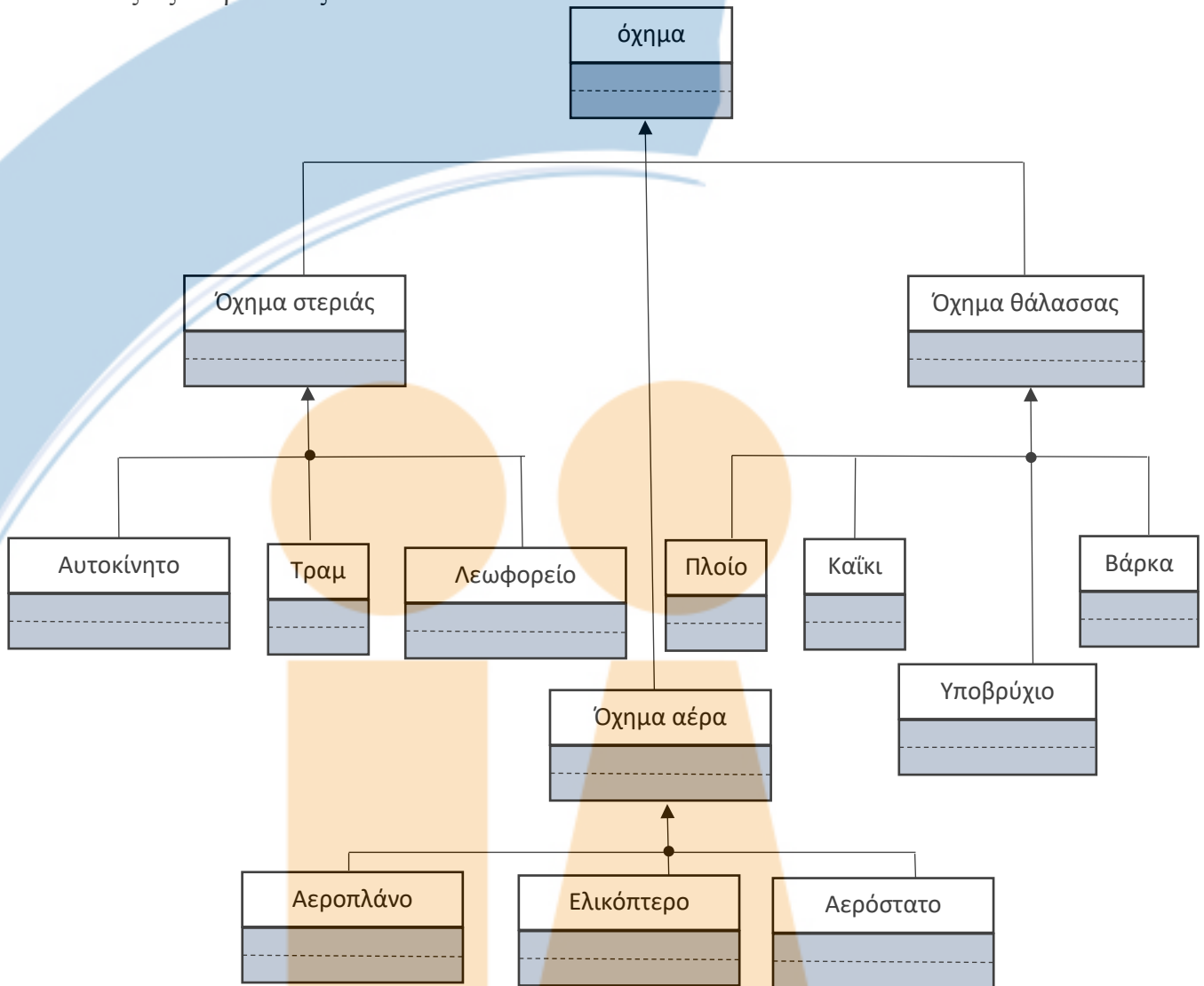
Ακολουθώντας, ο αριθμός 82 θα συγκριθεί με τον αριθμό 59. Και σε αυτή την περίπτωση έχει τιμή μεγαλύτερη, οπότε θα πάμε στο δεξί υποδένδρο του κόμβου 59.

Τελευταία σύγκριση του αριθμού 82 θα γίνει με τον αριθμό 65, όπου και πάλι είναι μεγαλύτερη η τιμή του 82 και θα εισαχθεί ως παιδί του πρώην φύλλου 65.

Το δένδρο θα λάβει τελικά τη μορφή:



**B4.** Οργανώνοντας τα σύνολα αντικειμένων σε διάγραμμα κληρονομικότητας κλάσεων θα έχουμε τη ακόλουθη διαγραμματική αναπαράσταση προσθέτοντας όπου χρειάζεται επιπλέον κλάσεις ως υπερκλάσεις:



**B5.** Α) εισαγωγή item σε ουρά QUEUE[N]

ΑΝ  $rear = N$  ΤΟΤΕ

    ΓΡΑΨΕ 'Γεμάτη ουρά'

ΑΛΛΙΩΣ

    ΑΝ  $front = 0$  ΚΑΙ  $rear = 0$  ΤΟΤΕ

$front \leftarrow 1$

    ΤΕΛΟΣ\_ΑΝ

$rear \leftarrow rear + 1$

$QUEUE[rear] \leftarrow item$

ΤΕΛΟΣ\_ΑΝ

B) εξαγωγή item από ουρά QUEUE[N]

**AN** front = 0 **ΚΑΙ** rear = 0 **TOTE**

**ΓΡΑΨΕ** 'Άδεια ουρά'

**ΑΛΛΙΩΣ**

**ΓΡΑΨΕ** QUEUE[front]

**AN** front = rear **TOTE**

front  $\leftarrow$  0

rear  $\leftarrow$  0

**ΑΛΛΙΩΣ**

front  $\leftarrow$  front + 1

**ΤΕΛΟΣ\_ΑΝ**

**ΤΕΛΟΣ\_ΑΝ**

Οπότε θα έχουμε:

1. rear = N
2. rear + 1
3. QUEUE[rear]
4. front = 0 **ΚΑΙ** rear = 0
5. front
6. 0
7. 0
8. front + 1



## ΘΕΜΑ Γ

**ΠΡΟΓΡΑΜΜΑ** Τετράποδοι\_Φίλοι  
**ΜΕΤΑΒΛΗΤΕΣ**

**ΑΚΕΡΑΙΕΣ:** ναι, όχι, , πλγυναϊκών, πλανδρών, πλανΔ, πλΑ, πλΒ, πλΓ, πλΔ, ηλικία, max

**ΠΡΑΓΜΑΤΙΚΕΣ:** ποσοστά

**ΧΑΡΑΚΤΗΡΕΣ:** απάντηση, φύλο, επιλογή

**ΑΡΧΗ**

ναι  $\leftarrow$  0

όχι  $\leftarrow$  0

πλΑ  $\leftarrow$  0

πλΒ  $\leftarrow$  0

πλΓ  $\leftarrow$  0

πλΔ  $\leftarrow$  0

πλγυναϊκών  $\leftarrow$  0

πλανδρών  $\leftarrow$  0

πλανΔ  $\leftarrow$  0

**ΑΡΧΗ\_ΕΠΑΝΑΛΗΨΗΣ**

**ΓΡΑΨΕ** 'Θέλετε να λάβετε μέρος στην έρευνα; Ναι / Όχι'

**ΔΙΑΒΑΣΕ** απάντηση

**ΑΝ** απάντηση = 'Ναι' **ΤΟΤΕ**

ναι  $\leftarrow$  ναι + 1

**ΔΙΑΒΑΣΕ** ηλικία

**ΑΡΧΗ\_ΕΠΑΝΑΛΗΨΗΣ**

**ΔΙΑΒΑΣΕ** φύλο

**ΜΕΧΡΙΣ\_ΟΤΟΥ** φύλο = 'Α' **Ή** φύλο = 'Γ'

**ΑΡΧΗ\_ΕΠΑΝΑΛΗΨΗΣ**

**ΔΙΑΒΑΣΕ** επιλογή

**ΜΕΧΡΙΣ\_ΟΤΟΥ** επιλογή = 'Α' **Ή** επιλογή = 'Β' **Ή** επιλογή = 'Γ' **Ή** &  
επιλογή = 'Δ'

**ΑΝ** επιλογή = 'Α' **ΤΟΤΕ**

πλΑ  $\leftarrow$  πλΑ + 1

**ΑΛΛΙΩΣ\_ΑΝ** επιλογή = 'Β' **ΤΟΤΕ**

πλΒ  $\leftarrow$  πλΒ + 1

**ΑΛΛΙΩΣ\_ΑΝ** επιλογή = 'Γ' **ΤΟΤΕ**

πλΓ  $\leftarrow$  πλΓ + 1

**ΑΛΛΙΩΣ**

πλΔ  $\leftarrow$  πλΔ + 1

**ΤΕΛΟΣ\_ΑΝ**

**ΑΝ** φύλο = 'Γ' **ΤΟΤΕ**

**ΑΝ** ηλικία  $\geq$  20 **ΚΑΙ** ηλικία  $\leq$  45 **ΤΟΤΕ**

**ΑΝ** επιλογή = 'Α' **ΤΟΤΕ**

πλγυναϊκών  $\leftarrow$  πλγυναϊκών + 1

**ΤΕΛΟΣ\_ΑΝ**

**ΤΕΛΟΣ\_ΑΝ**

**ΑΛΛΙΩΣ**

πλανδρών  $\leftarrow$  πλανδρών + 1

```
ΑΝ επιλογή = 'Δ' ΤΟΤΕ
    πλανΔ ← πλανΔ + 1
ΤΕΛΟΣ_ΑΝ
ΤΕΛΟΣ_ΑΝ
ΑΛΛΙΩΣ_ΑΝ απάντηση = 'Όχι' ΤΟΤΕ
    όχι ← όχι + 1
ΤΕΛΟΣ_ΑΝ
ΜΕΧΡΙΣ_ΟΤΟΥ ναι = 2500
max ← πλΑ
ΑΝ πλΒ > max ΤΟΤΕ
    max ← πλΒ
ΤΕΛΟΣ_ΑΝ
ΑΝ πλΓ > max ΤΟΤΕ
    max ← πλΓ
ΤΕΛΟΣ_ΑΝ
ΑΝ πλΔ > max ΤΟΤΕ
    max ← πλΔ
ΤΕΛΟΣ_ΑΝ
ΓΡΑΨΕ 'Η επικρατέστερη τοποθεσία για τη δημιουργία πάρκου είναι:', max
ΓΡΑΨΕ 'Ο συνολικός αριθμός των ατόμων που δεν απάντησαν είναι:', όχι
ΓΡΑΨΕ 'Οι προτιμήσεις για κάθε χώρο είναι: Α =', πλΑ, 'Β =', πλΒ, 'Γ =', πλΓ, 'Δ =', πλΔ
ΓΡΑΨΕ 'Οι γυναίκες 20 – 45 που προτιμούν τον χώρο Α είναι =', πλγυναικών
ΑΝ πλανδρών <> 0 ΤΟΤΕ
    ποσοστόαν ← πλανΔ / πλανδρών * 100
    ΓΡΑΨΕ 'Το ποσοστό ανδρών που θέλει το πάρκο στα όρια είναι:', ποσοστόαν, '%'
ΑΛΛΙΩΣ
    ΓΡΑΨΕ 'Δεν συμμετείχαν στην έρευνα καθόλου άνδρες'
ΤΕΛΟΣ_ΑΝ
ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ
```



## ΘΕΜΑ Δ

ΠΡΟΓΡΑΜΜΑ Διαπολιτισμική\_Εκπαίδευση

### ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: i, ΑΠΟΥΣΙΕΣ[250, 9], ΑΠΤΡΙ[250, 3], pos

ΧΑΡΑΚΤΗΡΕΣ: ΚΩΔ[250], κωδικός

### ΑΡΧΗ

ΚΑΛΕΣΕ ΕΙΣΑΓΩΓΗ(ΚΩΔ, ΑΠΟΥΣΙΕΣ)

ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 250

ΑΠΤΡΙ[i, 1] ← ΣΥΝΑΠ(i, ΑΠΟΥΣΙΕΣ, 1)

ΑΠΤΡΙ[i, 2] ← ΣΥΝΑΠ(i, ΑΠΟΥΣΙΕΣ, 4)

ΑΠΤΡΙ[i, 3] ← ΣΥΝΑΠ(i, ΑΠΟΥΣΙΕΣ, 7)

ΤΕΛΟΣ\_ΕΠΑΝΑΛΗΨΗΣ

ΔΙΑΒΑΣΕ κωδικός

ΟΣΟ κωδικός <> 'ΤΕΛΟΣ' ΕΠΑΝΑΛΑΒΕ

pos ← ΑΝΑΖ(κωδικός, ΚΩΔ)

ΑΝ pos <> 0 ΤΟΤΕ

ΑΝ ΑΠΟΥΣΙΕΣ[pos, 1] < 15 ΚΑΙ ΑΠΟΥΣΙΕΣ[pos, 2] < 15 ΚΑΙ &  
ΑΠΟΥΣΙΕΣ[pos, 3] < 15 ΤΟΤΕ

ΓΡΑΨΕ 'Δικαιούται συμμετοχής στις εξετάσεις'

ΑΛΛΙΩΣ

ΓΡΑΨΕ 'Δεν δικαιούται συμμετοχής στις εξετάσεις'

ΤΕΛΟΣ\_ΑΝ

ΑΛΛΙΩΣ

ΓΡΑΨΕ 'Δεν βρέθηκε ο κωδικός'

ΤΕΛΟΣ\_ΑΝ

ΔΙΑΒΑΣΕ κωδικός

ΤΕΛΟΣ\_ΕΠΑΝΑΛΗΨΗΣ

ΤΕΛΟΣ\_ΠΡΟΓΡΑΜΜΑΤΟΣ

!-----

ΣΥΝΑΡΤΗΣΗ ΣΥΝΑΠ(γρ, Α, μ): ΑΚΕΡΑΙΑ

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: γρ, Α[250, 9], μ

ΑΡΧΗ

ΣΥΝΑΠ ← Α[γρ, μ] + Α[γρ, μ + 1] + Α[γρ, μ + 2]

ΤΕΛΟΣ\_ΣΥΝΑΡΤΗΣΗΣ

!-----

**ΔΙΑΔΙΚΑΣΙΑ ΕΙΣΑΓΩΓΗ(K, B)**

**ΜΕΤΑΒΛΗΤΕΣ**

**ΧΑΡΑΚΤΗΡΕΣ:** K[250]

**ΑΚΕΡΑΙΕΣ:** i, j, B[250, 9]

**ΑΡΧΗ**

**ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 250**

**ΔΙΑΒΑΣΕ K[i]**

**ΓΙΑ j ΑΠΟ 1 ΜΕΧΡΙ 9**

**ΔΙΑΒΑΣΕ B[i, j]**

**ΤΕΛΟΣ\_ΕΠΑΝΑΛΗΨΗΣ**

**ΤΕΛΟΣ\_ΕΠΑΝΑΛΗΨΗΣ**

**ΤΕΛΟΣ\_ΔΙΑΔΙΚΑΣΙΑΣ**

!-----

**ΣΥΝΑΡΤΗΣΗ ΑΝΑΖ(key, K): ΑΚΕΡΑΙΑ**

**ΜΕΤΑΒΛΗΤΕΣ**

**ΑΚΕΡΑΙΕΣ:** i, pos

**ΧΑΡΑΚΤΗΡΕΣ:** key, K[250]

**ΛΟΓΙΚΕΣ:** flag

**ΑΡΧΗ**

flag  $\leftarrow$  ΨΕΥΔΗΣ

i  $\leftarrow$  1

pos  $\leftarrow$  0

**ΟΣΟ** i  $\leq$  250 **ΚΑΙ** flag = ΨΕΥΔΗΣ **ΕΠΑΝΑΛΑΒΕ**

**ΑΝ** key = K[i] **ΤΟΤΕ**

flag  $\leftarrow$  ΑΛΗΘΗΣ

pos  $\leftarrow$  i

**ΑΛΛΙΩΣ**

i  $\leftarrow$  i + 1

**ΤΕΛΟΣ\_ΑΝ**

**ΤΕΛΟΣ\_ΕΠΑΝΑΛΗΨΗΣ**

ΑΝΑΖ  $\leftarrow$  pos

**ΤΕΛΟΣ\_ΣΥΝΑΡΤΗΣΗΣ**