

Μάθημα 182

Τελευταίο μάθημα

Θέμα 1α

Χαρακτηρίστε καθεμία από τις παρακάτω προτάσεις ως σωστή (Σ) ή λανθασμένη (Λ):

1. Ο χαρακτήρας του κενού ανήκει στο αλφάβητο της γλώσσας. Σ
2. Σε μία αλφαριθμητική τιμή μπορεί να περιέχονται και ψηφία. Σ
3. Αν σε ένα δυαδικό αριθμό γίνει ολίσθηση προς τα δεξιά, τότε έχει πολλαπλασιαστεί επί δύο. ^{δαιρείται ακέραια διά 2} Λ
4. Η στοίβα χρόνου εκτέλεσης θα μπορούσε να περιέχει ταυτόχρονα δύο ή περισσότερες διευθύνσεις επιστροφής στο κύριο πρόγραμμα. Λ
5. Οι τελεστές DIV και MOD, δεν μπορούν να χρησιμοποιηθούν σε πραγματικούς αριθμούς. Σ
6. Μία λογική έκφραση που περιέχει μόνο αριθμητικά δεδομένα, περιλαμβάνει υποχρεωτικά συγκριτικό τελεστή. Σ
7. Μία λογική έκφραση περιέχει υποχρεωτικά λογικό τελεστή. Λ
8. Μία αλφαριθμητική μεταβλητή δεν μπορεί να λάβει τις τιμές "Αληθής" ή "Ψευδής", καθώς σε αυτή την περίπτωση θα αποτελεί λογική μεταβλητή. Λ
9. Ο μεταγλωττιστής εντοπίζει τα ^{συντακτικά} λογικά λάθη και εμφανίζει κατάλληλα διαγνωστικά μηνύματα. Λ
10. Ο αλγόριθμος της σειριακής ^{φυσικής} αναζήτησης ονομάζεται διαφορετικά και μέθοδος ευθείας ανταλλαγής. Λ
11. Η αναπαράσταση αλγορίθμου με φυσική γλώσσα κατά βήματα, μπορεί να παραβιάσει το κριτήριο της καθοριστικότητας. Σ

Θέμα 1β

Χαρακτηρίστε καθεμία από τις παρακάτω προτάσεις ως σωστή (Σ) ή λανθασμένη (Λ):

12. Ο βρόχος Για K από 0.5 μέχρι 5.5, εκτελείται 6 φορές. Σ
13. Αν $X \leftarrow K < 1$, τότε η X μπορεί να πάρει μόνο μία μεταξύ δύο συγκεκριμένων τιμών. Σ
14. Ορίσματα ονομάζονται αλλιώς οι τυπικές παράμετροι. Σ
15. Σε μία ουρά με ένα στοιχείο οι δείκτες εμπρός και πίσω έχουν την ίδια τιμή. Σ
16. Το πλήθος των στοιχείων μίας μη κενής ουράς, δίνεται από τον τύπο: $\text{rear} - \text{front} + 1$. Λ
17. Η συνθήκη “ΠΑΠΠΟΥΣ” > “ΠΑΠΟΥΣ” είναι ΨΕΥΔΗΣ. Λ
18. Μία κλάση A μπορεί να είναι έγκυρη υποκλάση της B αν έχει νόημα να πούμε «ένα B είναι ένα (is_a) A». Λ
ένα A είναι ένα (is_a) B
19. Στην εντολή εκχώρησης $X \leftarrow N / 2$ με N άρτιο ακέραιο, η μεταβλητή X είναι πάντα πραγματικού τύπου. Σ
20. Κάθε δένδρο γράφος είναι ειδική περίπτωση γράφου δένδρου. Λ
21. Υπάρχει η δυνατότητα να υλοποιηθούν ουρές και στοίβες στις οποίες να εφαρμόζεται η δυναμική παραχώρηση μνήμης. Σ
22. Τα Οι λίστες δένδρα είναι γραμμικές δομές δεδομένων. Λ

Θέμα 2α

Τι είναι η ενθυλάκωση και με τι μπορούμε να την παρομοιάσουμε;

Σε μια αντικειμενοστραφή εφαρμογή κάθε αντικείμενο αποτελεί ξεχωριστή οντότητα και περιέχει ενσωματωμένες τις ιδιότητες (δεδομένα) και τους κανόνες συμπεριφοράς του (μεθόδους).

Η δυνατότητα ενός αντικειμένου να συνδυάζει εσωτερικά τα δεδομένα και τις μεθόδους χειρισμού του καλείται ενθυλάκωση (encapsulation).

Την ενθυλάκωση μπορούμε να την παρομοιάσουμε σαν ένα κέλυφος που υπάρχει γύρω από κάθε αντικείμενο και διαχωρίζει τον εσωτερικό από τον εξωτερικό του κόσμο.

Τι είναι κλάση;

Κλάση είναι ο γενικός τύπος ενός αντικειμένου και καθορίζει τις αρχικές ιδιότητες και τη συμπεριφορά κάθε αντικειμένου που προέρχεται από αυτή.

Μια κλάση αποτελεί ένα αφαιρετικό στοιχείο και μπορεί να παράγει ένα απεριόριστο πλήθος **δομικά** ίδιων αντικειμένων.

Τι είναι η δομή δένδρο και ποιοι οι κανόνες του;

Ένα δένδρο είναι μία δομή που αποτελείται από ένα σύνολο κόμβων και ένα σύνολο ακμών μεταξύ των κόμβων με βάση τους εξής κανόνες:

- Υπάρχει ένας ξεχωριστός κόμβος που ονομάζεται ρίζα. Αυτός είναι ένας κόμβος χωρίς γονέα.
- Για κάθε κόμβο c , εκτός από τη ρίζα, υπάρχει μόνο μια ακμή που καταλήγει στον κόμβο αυτόν ξεκινώντας από κάποιον άλλον κόμβο p . Ο κόμβος p ονομάζεται γονέας του c και ο κόμβος c παιδί του p .
- Για κάθε κόμβο υπάρχει μία μοναδική διαδρομή, δηλαδή μια ακολουθία διαδοχικών ακμών, που ξεκινάει από τη ρίζα και τερματίζει σε αυτόν τον κόμβο.

Δένδρο θεωρούμε και το κενό δένδρο, δηλαδή το δένδρο που δεν έχει ούτε κόμβους, ούτε ακμές. Το κενό δένδρο είναι το μόνο δένδρο χωρίς ρίζα.

Θέμα 2β

Ποια τα πλεονεκτήματα των δυαδικών δένδρων αναζήτησης;

Τα δυαδικά δένδρα αναζήτησης συνδυάζουν τα πλεονεκτήματα των λιστών, όσον αφορά τις πράξεις της εισαγωγής και της διαγραφής, αλλά και τα πλεονεκτήματα των ταξινομημένων πινάκων, όσον αφορά την πράξη της αναζήτησης. Έχουμε λοιπόν τη δυνατότητα να αναζητήσουμε ένα στοιχείο το ίδιο γρήγορα όσο και σε έναν ταξινομημένο πίνακα αλλά και να εισαγάγουμε και να διαγράψουμε εύκολα ένα στοιχείο ακριβώς επειδή δουλεύουμε με δένδρα και όχι με πίνακες.

Αναφέρετε τις βασικές πράξεις των συνδεδεμένων λιστών:

- Εισαγωγή κόμβου στη λίστα.
- Διαγραφή κόμβου από τη λίστα.
- Έλεγχος για το αν η λίστα είναι κενή.
- Αναζήτηση κόμβου για την εύρεση συγκεκριμένου στοιχείου.
- Διάσχιση της λίστας και προσπέλαση των στοιχείων της.

Σε τι διαφέρει ένας γράφος από ένα δένδρο;

- Στα δένδρα έχουμε μόνο κατευθυνόμενες ακμές (μονόδρομες) ενώ στους γράφους οι ακμές μπορούν να έχουν διπλή κατεύθυνση.
- Ένας γράφος μπορεί να έχει κυκλικούς δεσμούς (σύνδεση κόμβου με τον εαυτό του), ενώ ένα δένδρο όχι.
- Τα μη κενά δένδρα έχουν κόμβο ρίζα, οι γράφοι όχι υποχρεωτικά.
- Στα δένδρα υπάρχει μοναδική διαδρομή που οδηγεί σε κάποιον κόμβο, στους γράφους μπορείς να καταλήξεις σε κάποιο κόμβο μέσω πολλών διαδρομών.

Θέμα 3

Να γραφούν οι παρακάτω μαθηματικές εκφράσεις σε ΓΛΩΣΣΑ:

α. $\sqrt{x - \sqrt{y}}$

$T_P(x - T_P(y))$

β. $\frac{|x| - \eta\mu\theta}{\sqrt{x^2 + 5}}$

$(A_T(x) - HM(\theta)) / T_P(x^2 + 5)$

γ. $2x + \frac{3(x+1)}{y^2 + 1} - e^x$

$2 * x + (3 * (x + 1)) / (y^2 + 1) - E(x)$

Θέμα 4

Να συμπληρώσετε τον παρακάτω πίνακα:

X	Z	(ΟΧΙ X) ΚΑΙ (ΟΧΙ Z)	(ΟΧΙ X ΚΑΙ Z) Η (X ΚΑΙ ΟΧΙ Z)
Ψευδής	Ψευδής	(ΟΧΙ Ψευδής) ΚΑΙ (ΟΧΙ Ψευδής) Αληθής ΚΑΙ Αληθής Αληθής	(ΟΧΙ Ψευδής ΚΑΙ Ψευδής) Η (Ψευδής ΚΑΙ ΟΧΙ Ψευδής) (Αληθής ΚΑΙ Ψευδής) Η (Ψευδής ΚΑΙ Αληθής) Ψευδής Η Ψευδής Ψευδής
Ψευδής	Αληθής	(ΟΧΙ Ψευδής) ΚΑΙ (ΟΧΙ Αληθής) Αληθής ΚΑΙ Ψευδής Ψευδής	(ΟΧΙ Ψευδής ΚΑΙ Αληθής) Η (Ψευδής ΚΑΙ ΟΧΙ Αληθής) (Αληθής ΚΑΙ Αληθής) Η (Ψευδής ΚΑΙ Ψευδής) Αληθής Η Ψευδής Αληθής
Αληθής	Ψευδής	(ΟΧΙ Αληθής) ΚΑΙ (ΟΧΙ Ψευδής) Ψευδής ΚΑΙ Αληθής Ψευδής	(ΟΧΙ Αληθής ΚΑΙ Ψευδής) Η (Αληθής ΚΑΙ ΟΧΙ Ψευδής) (Ψευδής ΚΑΙ Ψευδής) Η (Αληθής ΚΑΙ Αληθής) Ψευδής Η Αληθής Αληθής
Αληθής	Αληθής	(ΟΧΙ Αληθής) ΚΑΙ (ΟΧΙ Αληθής) Ψευδής ΚΑΙ Ψευδής Ψευδής	(ΟΧΙ Αληθής ΚΑΙ Αληθής) Η (Αληθής ΚΑΙ ΟΧΙ Αληθής) (Ψευδής ΚΑΙ Αληθής) Η (Ψευδής ΚΑΙ Ψευδής) Ψευδής Η Ψευδής Ψευδής

Θέμα 5α

Δίνεται το παρακάτω τμήμα αλγορίθμου:

$K \leftarrow 1$

ΓΙΑ i **ΑΠΟ** -1 **ΜΕΧΡΙ** -5 **ΜΕ_ΒΗΜΑ** -2

$K \leftarrow K * i$

ΓΡΑΨΕ K

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

Να μετατραπεί το τμήμα αυτό αλγορίθμου σε ισοδύναμο:

α. με χρήση της αλγοριθμικής δομής **ΟΣΟ**

$K \leftarrow 1$

$i \leftarrow -1$

ΟΣΟ $i \geq -5$ **ΕΠΑΝΑΛΑΒΕ**

$K \leftarrow K * i$

ΓΡΑΨΕ K

$i \leftarrow i - 2$

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

β. με χρήση της αλγοριθμικής δομής **ΜΕΧΡΙΣ_ΟΤΟΥ**

$K \leftarrow 1$

$i \leftarrow -1$

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ

$K \leftarrow K * i$

ΓΡΑΨΕ K

$i \leftarrow i - 2$

ΜΕΧΡΙΣ_ΟΤΟΥ $i < -5$

Θέμα 5β

Δίνεται το παρακάτω τμήμα αλγορίθμου:

ΔΙΑΒΑΣΕ Κ

$i \leftarrow K$

ΟΣΟ $i > -6$ ΕΠΑΝΑΛΑΒΕ

$i \leftarrow i - 2$

$K \leftarrow K * i$

ΓΡΑΨΕ Κ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

Να μετατραπεί το τμήμα αυτό αλγορίθμου σε ισοδύναμο:

α. με χρήση της αλγοριθμικής δομής **ΓΙΑ**

β. με χρήση της αλγοριθμικής δομής **ΜΕΧΡΙΣ_ΟΤΟΥ**

ΔΙΑΒΑΣΕ Κ

ΓΙΑ i ΑΠΟ Κ ΜΕΧΡΙ -5 ΜΕ_ΒΗΜΑ -2

$K \leftarrow K * (i - 2)$

ΓΡΑΨΕ Κ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΔΙΑΒΑΣΕ Κ

$i \leftarrow K$

ΑΝ $i > -6$ ΤΟΤΕ

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ

$i \leftarrow i - 2$

$K \leftarrow K * i$

ΓΡΑΨΕ Κ

ΜΕΧΡΙΣ_ΟΤΟΥ $i \leq -6$

ΤΕΛΟΣ_ΑΝ

Θέμα 6α

Δίνεται το παρακάτω τμήμα προγράμματος:

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ "ΔΩΣΕ ΘΕΤΙΚΟ ΑΡΙΘΜΟ"

ΔΙΑΒΑΣΕ**X**.....

ΜΕΧΡΙΣ_ΟΤΟΥ X> 0

ΓΙΑ i **ΑΠΟ** 0 **ΜΕΧΡΙ****X**..... **ΜΕ_ΒΗΜΑ****5**.....

 A ← i ^**2**.....

ΓΡΑΨΕ**A**.....

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

Να συμπληρωθεί τα κενά ώστε να υπολογίζει και να εμφανίζει τα τετράγωνα των πολλαπλασίων του 5 από το 0 μέχρι τον θετικό αριθμό X που διαβάστηκε.

Θέμα 6β

Δίνεται το παρακάτω τμήμα προγράμματος:

$S \leftarrow 0$

ΓΙΑ Χ ΑΠΟ **2** ΜΕΧΡΙ **20**

ΑΝ **$X \bmod 2 = 0$** ΤΟΤΕ

$S \leftarrow S + X^{(X-1)}$

ΑΛΛΙΩΣ

$S \leftarrow S - X^{(X+1)}$

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

Να συμπληρωθούν τα κενά, ώστε να υπολογίζεται το άθροισμα: $2^1 - 3^4 + 4^3 - 5^6 + \dots - 19^{20} + 20^{19}$

Θέμα 6γ

Δίνεται το παρακάτω τμήμα προγράμματος:

$S \leftarrow 0$

ΓΙΑ Κ ΑΠΟ **1** ΜΕΧΡΙ **21**

$A \leftarrow \dots \mathbf{2 * K - 12} \dots$

$S \leftarrow S + \dots \mathbf{A} \dots$

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ **S**

$$\alpha * 1 + \beta = -10 \quad [1]$$

$$\alpha * 2 + \beta = -8 \quad [2]$$

$$[2] - [1] \rightarrow \alpha = -8 - (-10)$$

$$\rightarrow \alpha = 2$$

$$[1] \rightarrow 2 + \beta = -10$$

$$\rightarrow \beta = -12$$

$$2 * K - 12 = 30 \rightarrow$$

$$2 * K = 42 \rightarrow$$

$$K = 21$$

Να συμπληρωθούν τα κενά, ώστε να υπολογίζεται και να εμφανίζεται το άθροισμα: $-10 - 8 - 6 - \dots + 28 + 30$

Θέμα 6δ

Παρακάτω δίνεται το τμήμα προγράμματος που δημιουργεί τον πίνακα Π και το οποίο περιέχει πέντε κενά που πρέπει να συμπληρωθούν κατάλληλα.

ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 5

ΓΙΑ j ΑΠΟ 1 ΜΕΧΡΙ 5

ΑΝ i = j ΤΟΤΕ

Π[i, j] ← i ^ 2

ΑΛΛΙΩΣ_ΑΝ i < j ΤΟΤΕ

Π[i, j] ← i + j

ΑΛΛΙΩΣ

Π[i, j] ← i * j

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

Π

1	3	4	5	6
2	4	5	6	7
3	6	9	7	8
4	8	12	16	9
5	10	15	20	25

Να συμπληρωθούν τα κενά, ώστε να παράγεται ο παραπάνω πίνακας Π.

Θέμα 7 / σελίδα (1 / 2)

Μία εφαρμογή υπολογίζει τη μηνιαία έκπτωση (σε ποσοστό) ενός πελάτη σε ένα e-shop με βάση τον αριθμό των προϊόντων που αγόρασε τον προηγούμενο μήνα. Οι κανόνες είναι:

Αν δεν αγοράσει κάτι ή αν αγοράσει έως και 4 προϊόντα εμφανίζει το μήνυμα "ΕΚΠΤΩΣΗ 0%"

Διαφορετικά αν αγοράσει πάνω από 4 και έως και 15 προϊόντα εμφανίζει το μήνυμα "ΕΚΠΤΩΣΗ 5%".

Διαφορετικά αν αγοράσει πάνω από 15 και έως και 50 προϊόντα εμφανίζει το μήνυμα "ΕΚΠΤΩΣΗ 10%".

Σε οποιαδήποτε άλλη περίπτωση εμφανίζει το μήνυμα "ΕΚΤΟΣ ΟΡΙΩΝ".

Σχεδιάστε τα κατάλληλα σενάρια ελέγχου (test cases) χρησιμοποιώντας τη στρατηγική μαύρου κουτιού.

Βήμα 1ο: Δημιουργία ισοδύναμων διαστημάτων

Από την εκφώνηση προκύπτουν τρία διαστήματα για την είσοδο:

- $0 \leq X \leq 4$
- $4 < X \leq 15$ και
- $15 < X \leq 50$ και

Επίσης υπάρχουν δύο διαστήματα μη έγκυρων τιμών εισόδου:

- $X < 0$ και και $X > 50$



Θέμα 7 / σελίδα (2 / 2)

Βήμα 2ο: Καθορισμός ακραίων τιμών διαστημάτων



Στο προηγούμενο διάγραμμα φαίνεται ότι λείπουν κάποια άκρα. Για να τα υπολογίσουμε αρκεί να προσθέσουμε ή να αφαιρέσουμε 1 από το άκρο του προηγούμενου ή επόμενου διαστήματος αντίστοιχα, αφού σύμφωνα με την εκφώνηση το πλήθος των προϊόντων είναι ένας ακέραιος αριθμός.

Με αυτό τον τρόπο καταλήγουμε στο παρακάτω διάγραμμα:



Βήμα 3ο: Δημιουργία σεναρίων ελέγχου

A/A	Είσοδος	Αναμενόμενο Αποτέλεσμα	Περίπτωση που ελέγχεται
1	-1	ΕΚΤΟΣ ΟΡΙΩΝ	Άνω άκρο διαστήματος: $X < 0$
2	0	ΕΚΠΤΩΣΗ 0%	Κάτω άκρο διαστήματος: $0 \leq X \leq 4$
3	4	ΕΚΠΤΩΣΗ 0%	Άνω άκρο διαστήματος: $0 \leq X \leq 4$
4	5	ΕΚΠΤΩΣΗ 5%	Κάτω άκρο διαστήματος: $4 < X \leq 15$
5	15	ΕΚΠΤΩΣΗ 5%	Άνω άκρο διαστήματος: $4 < X \leq 15$
6	16	ΕΚΠΤΩΣΗ 10%	Κάτω άκρο διαστήματος: $15 < X \leq 50$
7	50	ΕΚΠΤΩΣΗ 10%	Άνω άκρο διαστήματος: $15 < X \leq 50$
8	51	ΕΚΤΟΣ ΟΡΙΩΝ	Κάτω άκρο διαστήματος: $X > 50$

Θέμα 8α

Δίνεται το παρακάτω τμήμα αλγορίθμου:

$X \leftarrow 0$

$A \leftarrow 10$

$B \leftarrow 14$

ΟΣΟ $B > 0$ **ΕΠΑΝΑΛΑΒΕ**

ΑΝ $B \bmod 2 = 1$ **ΤΟΤΕ**

$X \leftarrow X + A$

ΤΕΛΟΣ_ΑΝ

$A \leftarrow A * 2$

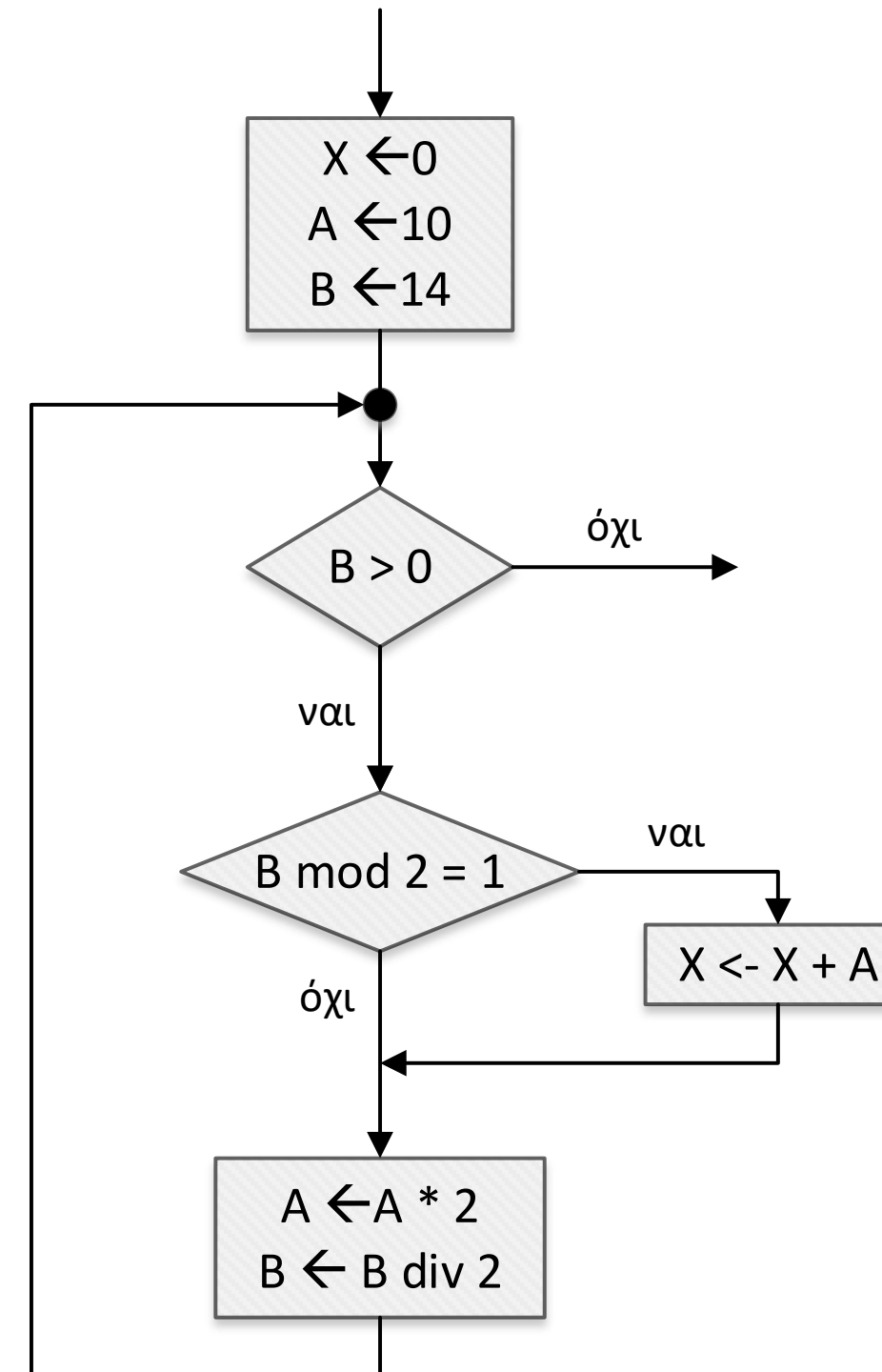
$B \leftarrow B \text{ div } 2$

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

α. Να γραφούν οι τιμές των μεταβλητών X, A και B στο τέλος κάθε επανάληψης κατά την εκτέλεση του παραπάνω αλγόριθμου.

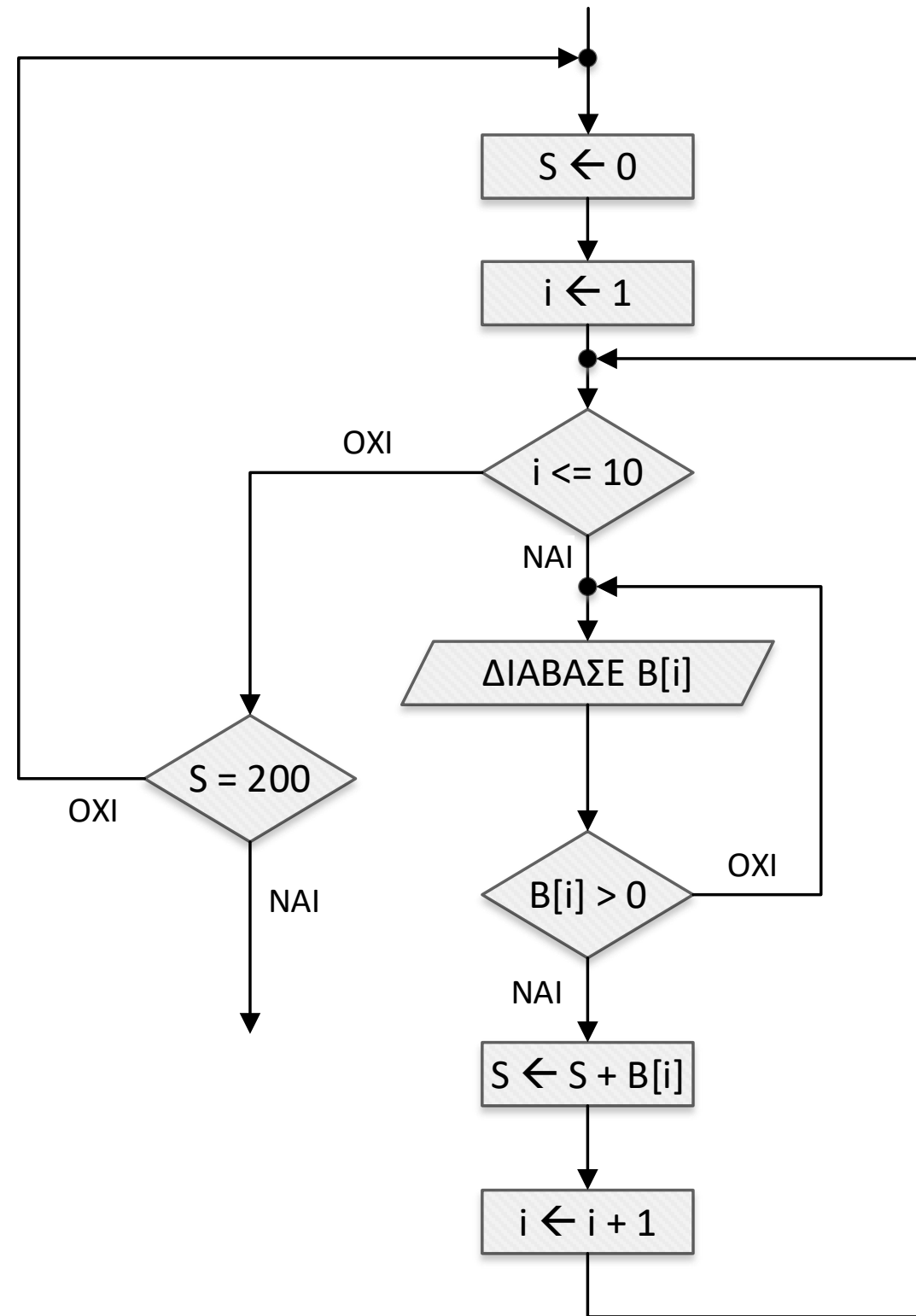
$B > 0$	$B \bmod 2 = 1$	X	A	B
		0	10	14
Αληθής	Ψευδής	0	20	7
Αληθής	Αληθής	20	40	3
Αληθής	Αληθής	60	80	1
Αληθής	Αληθής	140	160	0
Ψευδής				

β. Να σχεδιαστεί το αντίστοιχο διάγραμμα ροής.



Θέμα 8β

Για το παρακάτω διάγραμμα ροής να γραφεί το αντίστοιχο τμήμα κωδικοποίησης:



```
ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ
S ← 0
i ← 1
ΟΣΟ i ≤ 10 ΕΠΑΝΑΛΑΒΕ
  ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ
  ΔΙΑΒΑΣΕ B[i]
  ΜΕΧΡΙΣ_ΟΤΟΥ B[i] > 0
  S ← S + B[i]
  i ← i + 1
  ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΜΕΧΡΙΣ_ΟΤΟΥ S = 200
```

```
ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ
S ← 0
ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 10
  ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ
  ΔΙΑΒΑΣΕ B[i]
  ΜΕΧΡΙΣ_ΟΤΟΥ B[i] > 0
  S ← S + B[i]
  ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΜΕΧΡΙΣ_ΟΤΟΥ S = 200
```

Θέμα 9

Το παρακάτω τμήμα αλγορίθμου προσπαθεί να διαβάσει 50 αριθμούς, να υπολογίσει και να εμφανίσει το άθροισμα και τον μέσο όρο τους.

1. Για i από 101 μέχρις 150
2. Διάβασε X
3. $S \leftarrow S + X$
4. Τέλος επανάληψης
5. $MO \leftarrow S / (150 - 101) + 1$
6. Εμφάνισε S, MO

0. $S \leftarrow 0$
1. Για i από 101 μέχρι 150
2. Διάβασε X
3. $S \leftarrow S + X$
4. Τέλος_επανάληψης
5. $MO \leftarrow S / (150 - 101 + 1)$
6. Εμφάνισε S, MO

Να σημειωθούν οι γραμμές στις οποίες υπάρχουν λάθη και να χαρακτηριστούν ως λογικά ή συντακτικά ή χρόνου εκτέλεσης.

- Στη γραμμή 1 υπάρχει συντακτικό λάθος (μέχρις).
- Στη γραμμή 3 υπάρχει λάθος χρόνου εκτέλεσης (πρέπει πριν τη ΓΙΑ να αρχικοποιήσουμε το $S \leftarrow 0$).
- Στη γραμμή 4 υπάρχει συντακτικό λάθος (λείπει η κάτω παύλα ανάμεσα στις δύο λέξεις).
- Στη γραμμή 5 υπάρχει λογικό λάθος (πρώτα γίνεται διαίρεση με το 49 και μετά προστίθεται το 1).

Θέμα 10

Δίνεται το παρακάτω τμήμα αλγόριθμου, που υλοποιεί την πρώτη φάση της συγχώνευσης των ταξινομημένων πινάκων A[100] και B[200] σε πίνακα Γ[300].

Ο πίνακας A είναι ταξινομημένος σε αύξουσα σειρά και ο πίνακας B σε φθίνουσα.

Το τμήμα αυτό επεξεργάζεται τους πίνακες A και B τοποθετώντας τα στοιχεία τους στον πίνακα Γ σε αύξουσα σειρά.

Η διαδικασία σταματά, όταν εξαντληθούν τα στοιχεία ενός από τους πίνακες A και B.

Συμπληρώστε σε κάθε μία από τις παρακάτω υλοποιήσεις, κάθε κενό με τον τελεστή ή τη μεταβλητή που αντιστοιχεί.

$i \leftarrow \dots$

$j \leftarrow \dots$

$k \leftarrow 1$

Όσο $i \leq 100$ και $j \geq 1$ επανάλαβε

Αν $A[i] \leq B[j]$ τότε

$\Gamma[k] \leftarrow A[i]$

$i \leftarrow i + 1$

Αλλιώς

$\Gamma[k] \leftarrow B[j]$

$j \leftarrow j - 1$

Τέλος_αν

$k \leftarrow k + 1$

Τέλος_επανάληψης

Θέμα 11

Για την ταξινόμηση, σε φθίνουσα σειρά, των στοιχείων ενός μονοδιάστατου πίνακα αριθμών $\Pi[30]$ μπορεί να ακολουθηθεί η παρακάτω διαδικασία:

Αρχικά, ο πίνακας σαρώνεται από την αρχή μέχρι το τέλος του, προκειμένου να βρεθεί το μεγαλύτερο στοιχείο του. Αυτό το στοιχείο τοποθετείται στην αρχή του πίνακα, ανταλλάσσοντας θέσεις με το στοιχείο της πρώτης θέσης του πίνακα. Η σάρωση του πίνακα επαναλαμβάνεται, ξεκινώντας τώρα από το δεύτερο στοιχείο του πίνακα. Το μεγαλύτερο από τα στοιχεία που απέμειναν ανταλλάσσει θέσεις με το στοιχείο της δεύτερης θέσης του πίνακα. Η σάρωση επαναλαμβάνεται, ξεκινώντας από το τρίτο στοιχείο του πίνακα, μετά από το τέταρτο στοιχείο του πίνακα κ.ο.κ.

Συμπληρώστε τα κενά στο παρακάτω τμήμα αλγορίθμου, ώστε να επιτελεί τη λειτουργία που περιγράφηκε:

Για k από 1 μέχρι 29

$\theta \leftarrow \dots^k\dots$

Για i από k μέχρι 30

Αν $\Pi[i] \dots^>\dots \Pi[\theta]$ τότε

$\theta \leftarrow \dots^i\dots$

Τέλος_αν

Τέλος_επανάληψης

αντιμετάθεσε $\Pi[k]$, $\Pi[\theta]$

Τέλος_επανάληψης

Θέμα 12

Ο παρακάτω αλγόριθμος αντιγράφει τα στοιχεία ενός μονοδιάστατου πίνακα $A[Y]$, όπου $Y = M * N$, σε δισδιάστατο πίνακα $B[M, N]$ ξεκινώντας από την πρώτη στήλη και συνεχίζοντας με κάθε επόμενη στήλη γεμίζοντας καθεμιά από πάνω προς τα κάτω:

Συμπληρώστε τα κενά, ώστε ο αλγόριθμος να επιτελεί τη λειτουργία που περιγράφεται.

Αλγόριθμος Αντιγραφή

Δεδομένα // A, M, N //

$k \leftarrow 0$

Για ... i ... από 1 μέχρι ... N ...

 Για ... j ... από 1 μέχρι ... M ...

$k \leftarrow k + 1$

$B[j, i] \leftarrow A[...k.]$

 Τέλος_Επανάληψης

Τέλος_Επανάληψης

Αποτελέσματα // B //

Τέλος Αντιγραφή

Θέμα 13

(ΣΑΒΒΑΣ) Ο διπλανός αλγόριθμος λαμβάνει στην είσοδό του μία ουρά που υλοποιείται με τον πίνακα $Q[15]$, καθώς και τις τιμές των δεικτών του $front$ και $rear$ και μέσω εξαγωγής των στοιχείων αυτής δημιουργεί και εμφανίζει τον πίνακα $A[4, 5]$.

Συγκεκριμένα, κάθε συνεχόμενη τριάδα στοιχείων της ουράς περιέχει στην 1^η θέση τη γραμμή i , στη 2^η θέση τη στήλη j και στην 3^η θέση την τιμή $A[i, j]$. Όλα τα υπόλοιπα στοιχεία του A είναι μηδενικά.

Συμπληρώστε τα κενά του αλγορίθμου ώστε να συμβεί αυτό που περιγράφηκε προηγουμένως.

Τι θα εμφανιστεί κατά την έξοδο, εάν η ουρά είναι η:

			1	2	10	2	1	20	2	5	30	3	3	-10
--	--	--	---	---	----	---	---	----	---	---	----	---	---	-----

και οι δείκτες που δίνονται κατά την είσοδο είναι $front = 4$ και $rear = 15$;

0	10	0	0	0
20	0	0	0	30
0	0	-10	0	0
0	0	0	0	0

Αλγόριθμος Αραιός

Δεδομένα // $Q, front, rear$ //

Για i από 1 μέχρι 5

Για j από 1 μέχρι 4

$A[\dots j \dots, \dots i \dots] \leftarrow 0$

Τέλος_επανάληψης

Τέλος_επανάληψης

Για k από $front$ μέχρι $rear$

$item \leftarrow Q[front]$

$front \leftarrow front + 1$

Αν $k \bmod 3 = 1$ τότε

$i \leftarrow item$

Αλλιώς_αν $k \bmod 3 = 2$ τότε

$j \leftarrow item$

Αλλιώς

$A[i, j] \leftarrow item$

Τέλος_αν

Τέλος_επανάληψης

Για i από 1 μέχρι 4

Εμφάνισε $A[i,1], A[i,2], A[i,3], A[i,4], A[i,5]$

Τέλος_επανάληψης

Τέλος Αραιός

Θέμα 14

Δίνεται το παρακάτω πρόγραμμα και υποπρογράμματα: a, b, x

ΠΡΟΓΡΑΜΜΑ Κλήση_Υποπρογραμμάτων
ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: a, b, x

ΑΡΧΗ

$a \leftarrow 1$

$b \leftarrow 2$

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ

ΑΝ $a \leq 4$ **ΤΟΤΕ**

ΚΑΛΕΣΕ Διαδ1(a, b, x)

ΑΛΛΙΩΣ

$x \leftarrow \text{Συν1}(a, b)$

ΤΕΛΟΣ_ΑΝ

ΓΡΑΨΕ a, b, x

ΜΕΧΡΙΣ_ΟΤΟΥ $x > 11$

ΓΡΑΨΕ x

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

ΔΙΑΔΙΚΑΣΙΑ Διαδ1 (λ, κ, μ)

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: κ, λ, μ

ΑΡΧΗ

$\kappa \leftarrow \kappa + 1$ $b \leftarrow b + 1$

$\lambda \leftarrow \lambda + 3$ $a \leftarrow a + 3$

$\mu \leftarrow \kappa + \lambda$ $x \leftarrow b + a$

ΤΕΛΟΣ_ΔΙΑΔΙΚΑΣΙΑΣ

ΣΥΝΑΡΤΗΣΗ Συν1(c, z): **ΑΚΕΡΑΙΑ**

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: c, z

ΑΡΧΗ

$z \leftarrow z + 2$

$c \leftarrow c * 2$

$\text{Συν1} \leftarrow c + z$

ΤΕΛΟΣ_ΣΥΝΑΡΤΗΣΗΣ

ΠΡΟΓΡΑΜΜΑ Κλήση_Υποπρογραμμάτων

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: $a, b, x, c, z, \text{Συν1}$

ΑΡΧΗ

$a \leftarrow 1$

$b \leftarrow 2$

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ

ΑΝ $a \leq 4$ **ΤΟΤΕ**

$b \leftarrow b + 1$

$a \leftarrow a + 3$

$x \leftarrow b + a$

ΑΛΛΙΩΣ

$c \leftarrow a$

$z \leftarrow b$

$z \leftarrow z + 2$

$c \leftarrow c * 2$

$\text{Συν1} \leftarrow c + z$

$x \leftarrow \text{Συν1}$

ΤΕΛΟΣ_ΑΝ

ΓΡΑΨΕ a, b, x

ΜΕΧΡΙΣ_ΟΤΟΥ $x > 11$

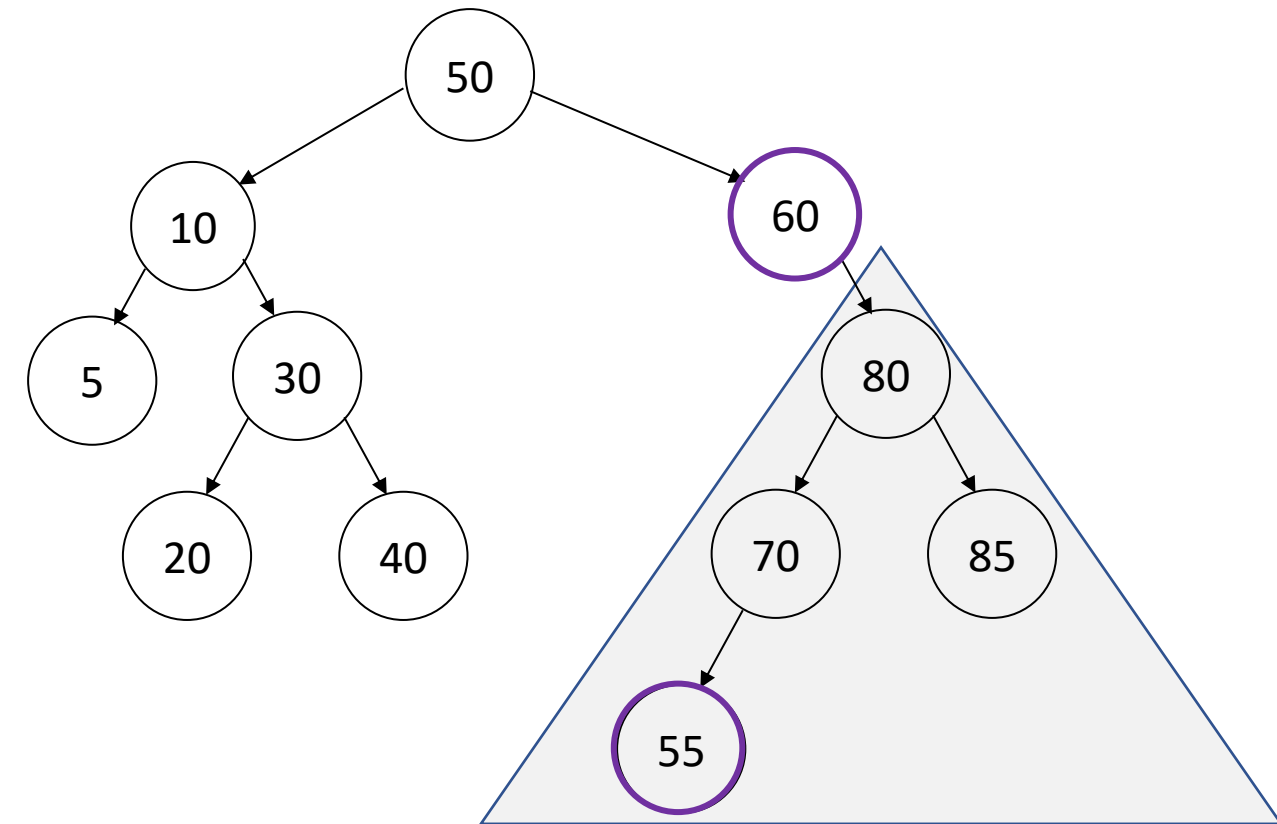
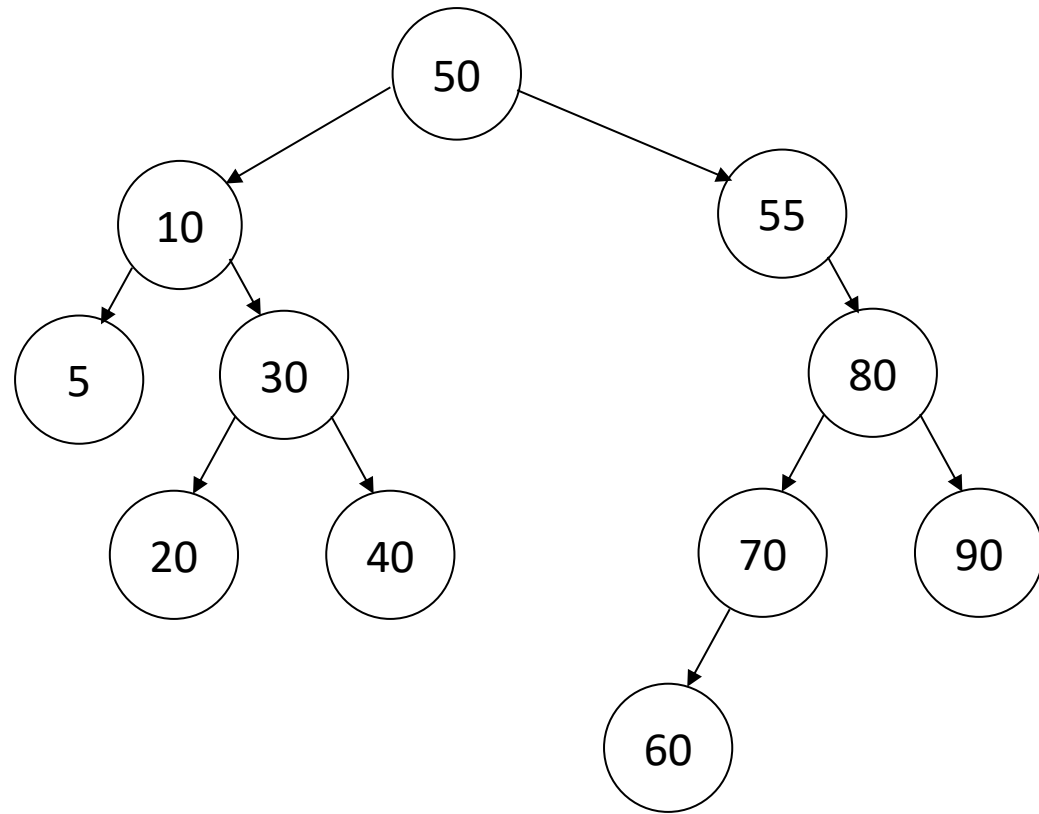
ΓΡΑΨΕ x

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

Να γράψετε στο τετράδιό σας το αντίστοιχο πρόγραμμα, χωρίς τη χρήση των υποπρογραμμάτων.

Θέμα 15α

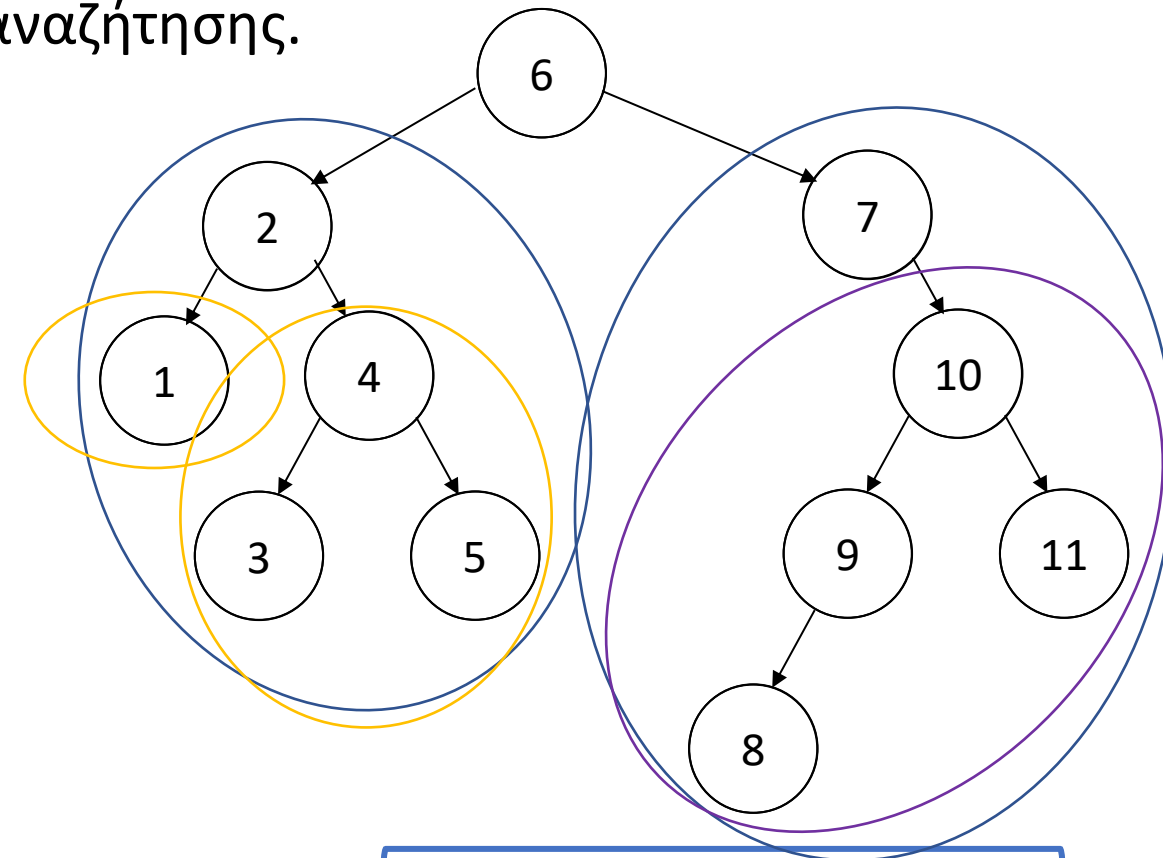
Ποιο από τα παρακάτω ΔΕΝ είναι δυαδικό δένδρο αναζήτησης και γιατί;



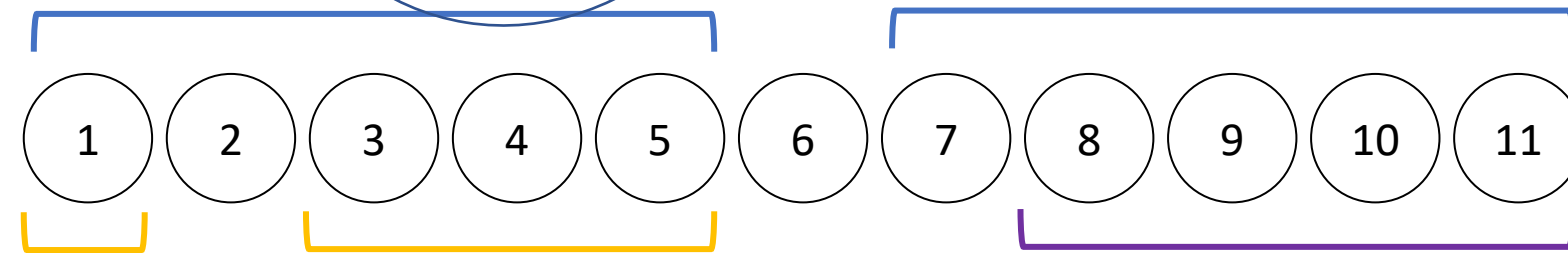
Δεν είναι το 2^ο, καθώς ο κόμβος με τιμή 55, βρίσκεται στο δεξί υποδένδρο του κόμβου με τιμή 60, οπότε και θα έπρεπε να περιέχει τιμή μεγαλύτερη του 60.

Θέμα 15β

Στους κόμβους του παρακάτω δένδρου να τοποθετηθούν οι ακέραιοι αριθμοί 1, 2, 3, ..., 11, στις θέσεις που πρέπει, ώστε να καταστεί δυαδικό δένδρο αναζήτησης.



Τοποθετούμε τους αριθμούς σε αύξουσα σειρά:



Η ρίζα έχει αριστερά της 5 κόμβους, δηλαδή τις 5 μικρότερες τιμές. Οπότε η ρίζα περιέχει την 6^η στη σειρά τιμή.

Οι υπόλοιπες τιμές χωρίζονται σε δύο ομάδες, που αντιστοιχούν στα δύο υποδένδρα της ρίζας.

Συνεχίζουμε με αντίστοιχο τρόπο σε κάθε ένα από τα δύο υποδένδρα της ρίζας. Για παράδειγμα:

Στο αριστερό υποδένδρο, η ρίζα έχει αριστερά της 1 κόμβο, δηλαδή την 1^η στη σειρά τιμή της αριστερής ομάδας, άρα η ρίζα του υποδένδρου θα έχει τη 2^η τιμή.

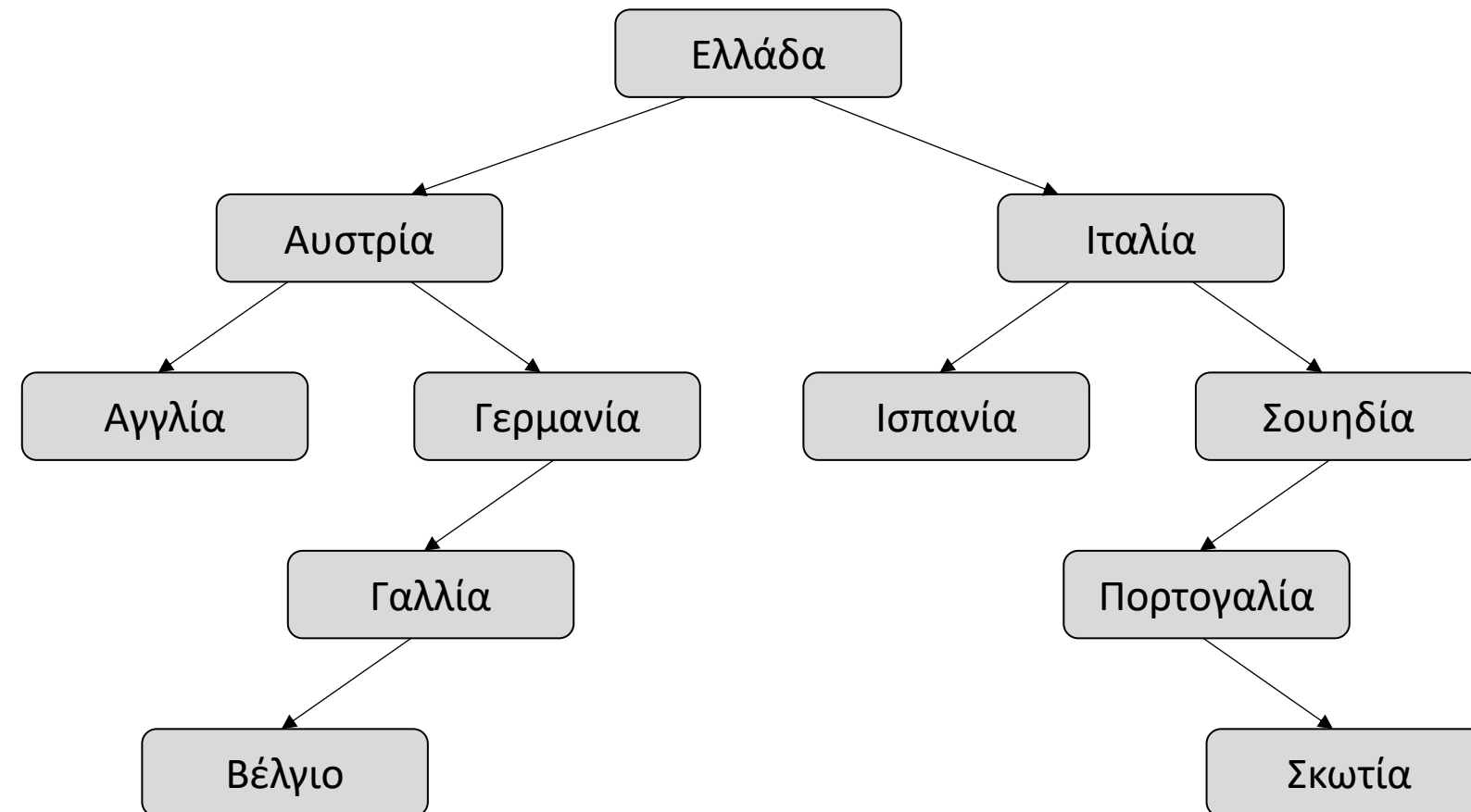
Στο δεξί υποδένδρο, η ρίζα δεν έχει κόμβο αριστερά της, οπότε αυτή έχει τη μικρότερη τιμή της δεξιάς ομάδας.

Και ούτω καθεξής.

Θέμα 15γ

(ΣΑΒΒΑΣ) Δημιουργήστε με τις παρακάτω Ευρωπαϊκές χώρες και με τη σειρά που δίνονται, ένα δυαδικό δένδρο αναζήτησης:

Ελλάδα , Ιταλία , Ισπανία , Αυστρία , Γερμανία , Σουηδία , Πορτογαλία , Σκωτία , Γαλλία , Αγγλία , Βέλγιο.



Θέμα 15δ

(ΣΑΒΒΑΣ) Έστω πως οι κόμβοι ενός δένδρου έχουν τοποθετηθεί σε διαδοχικές θέσεις μνήμης με διευθύνσεις από 1 μέχρι και 12, με τη ρίζα να βρίσκεται στη διεύθυνση 11.

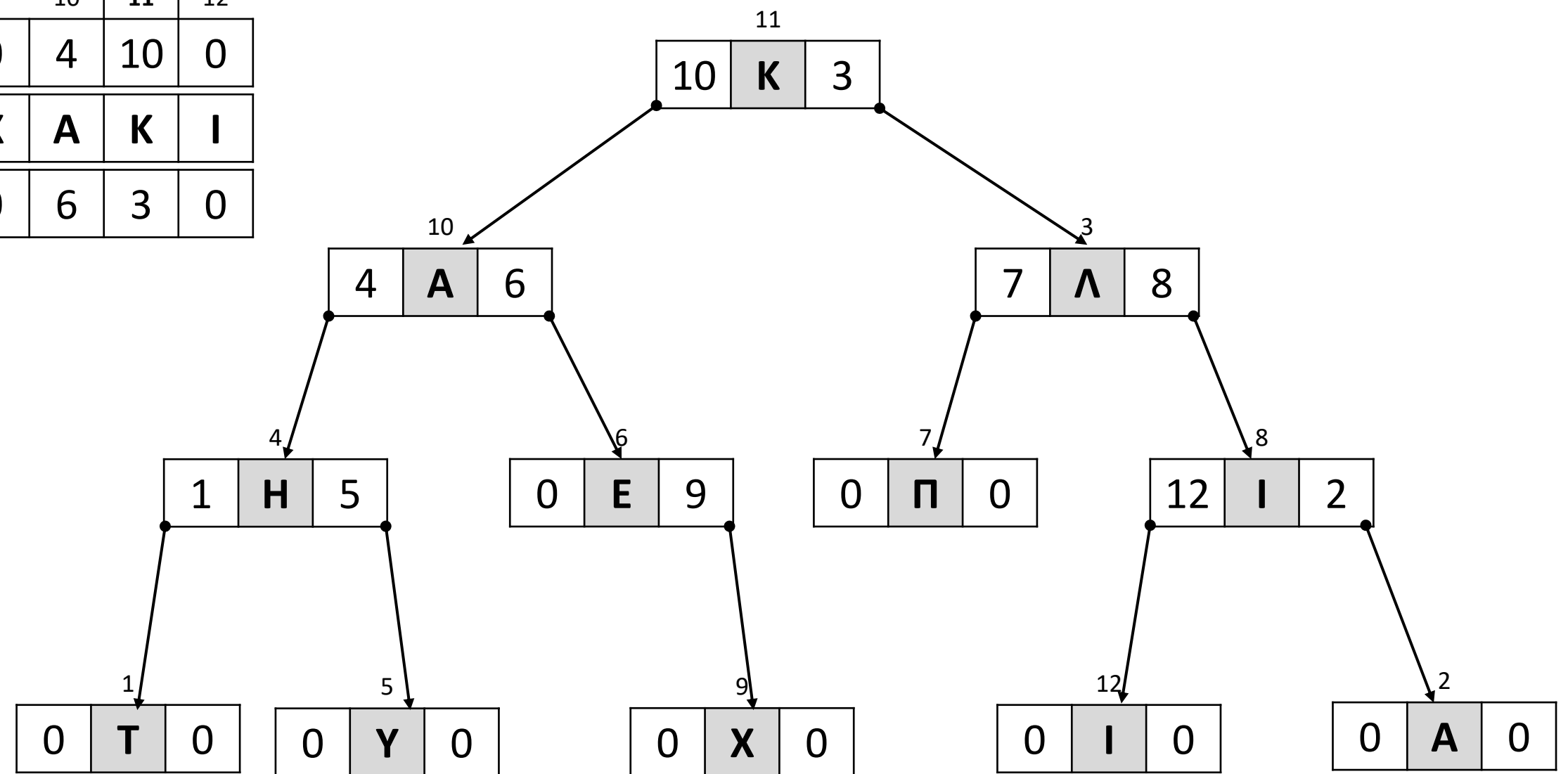
Κάθε κόμβος του δένδρου αποτελείται από 3 πεδία. Το 1^ο και το 3^ο πεδίο περιέχουν τις διευθύνσεις του αριστερού και του δεξιού, αντίστοιχα παιδιού του κόμβου, ενώ το 2^ο πεδίο περιέχει το δεδομένο του κόμβου.

Εάν κάποιο από το 1^ο ή το 3^ο πεδίο περιέχει την τιμή 0, σημαίνει πως ο κόμβος δεν έχει αριστερό ή δεξί παιδί αντίστοιχα.

Οι πληροφορίες για τη δομή αυτή βρίσκονται στους παράλληλους πίνακες που ακολουθούν:

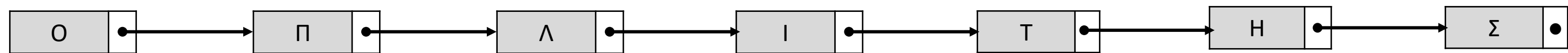
	1	2	3	4	5	6	7	8	9	10	11	12
1 ^ο πεδίο	0	0	7	1	0	0	0	12	0	4	10	0
2 ^ο πεδίο	Τ	Α	Λ	Η	Υ	Ε	Π	Ι	Χ	Α	Κ	Ι
3 ^ο πεδίο	0	0	8	5	0	9	0	2	0	6	3	0

Σχεδιάστε το δένδρο.

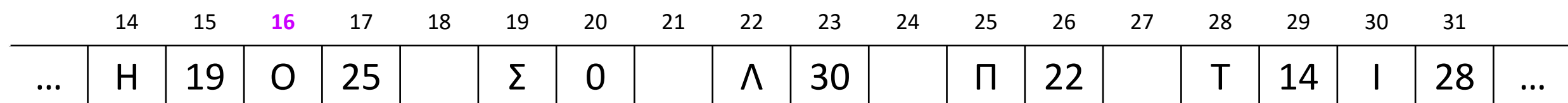


Θέμα 16

(ΣΑΒΒΑΣ) Δίνεται μια λίστα η οποία αποτελείται από 7 κόμβους της οποίας η κεφαλή περιέχει την τιμή 16. Το πρώτο πεδίο του κάθε κόμβου είναι ένα γράμμα και το δεύτερο πεδίο είναι η διεύθυνση του επόμενου κόμβου, όπως φαίνεται στο παρακάτω διάγραμμα, που σχηματίζει τη λέξη ΟΠΛΙΤΗΣ.

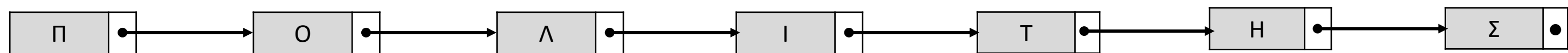


Η λίστα αυτή απεικονίζεται στη μνήμη με τη μορφή που φαίνεται στο παρακάτω σχήμα:

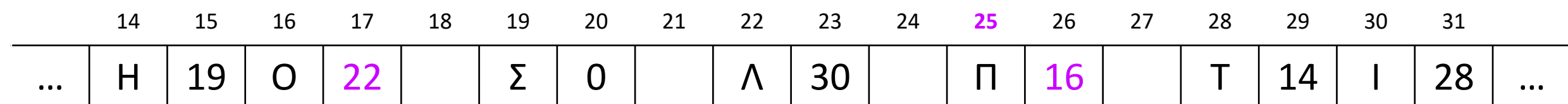


Στον τελευταίο κόμβο, το δεύτερο πεδίο έχει την τιμή 0, η οποία σηματοδοτεί το τέλος της λίστας.

Να σχεδιάσετε την απεικόνιση της μνήμης και να αναφέρετε την τιμή που περιέχει ο κόμβος κορυφή, μετά από την αντιμετάθεση του κατάλληλου ζεύγους κόμβων στην αρχική λίστα, ώστε να σχηματιστεί η λέξη ΠΟΛΙΤΗΣ.



Η κεφαλή θα περιέχει τη διεύθυνση του κόμβου 'Π', δηλαδή την τιμή 25, ο δείκτης του κόμβου 'Π' θα είναι η διεύθυνση του κόμβου 'Ο', δηλαδή 16, ο δείκτης του κόμβου 'Ο' θα είναι η διεύθυνση του κόμβου 'Λ', δηλαδή 22, οπότε η μνήμη θα έχει τη μορφή:



Θέμα 17

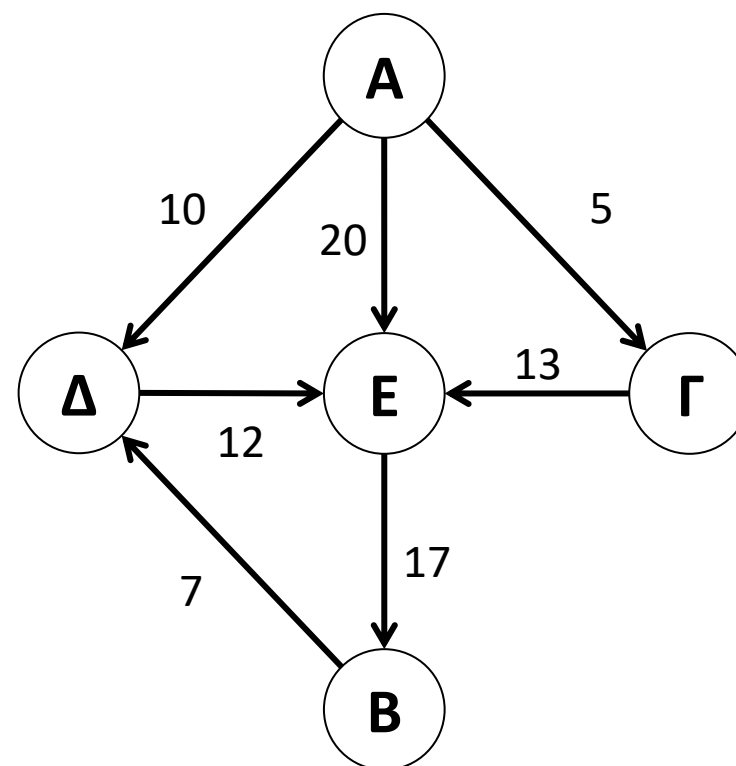
Ο πίνακας που ακολουθεί αναφέρει τους χρόνους μετάβασης, άρα και τις γειτνιάσεις μεταξύ πέντε κόμβων.

Εάν για παράδειγμα στη 1η γραμμή και στην 4η στήλη υπάρχει το 10, σημαίνει πως από τον κόμβο Α μπορούμε να μεταβούμε στον κόμβο Δ, σε 10 λεπτά.

Εάν δεν υπάρχει τρόπος μετάβασης από έναν κόμβο σε άλλον, στην αντίστοιχη θέση του πίνακα έχουμε την τιμή 0.

Με βάσει τον πίνακα αυτό, σχεδιάστε τον αντίστοιχο γράφο και βρείτε το μικρότερο χρόνο μετάβασης από τον Α στον Ε.

	A	B	Γ	Δ	E
A	0	0	5	10	20
B	0	0	0	7	0
Γ	0	0	0	0	13
Δ	0	0	0	0	12
E	0	17	0	0	0



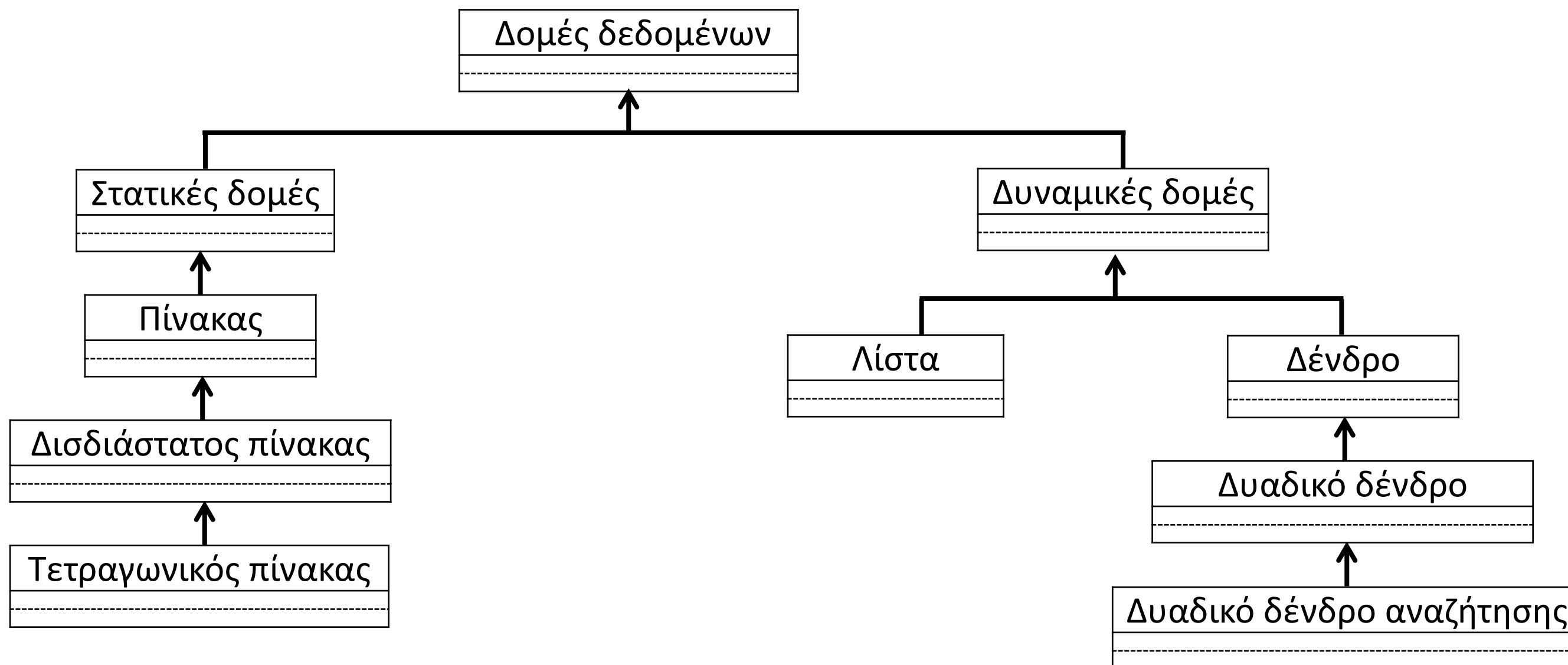
$$A \xrightarrow{20} E \xrightarrow{\text{σύνολο}} \mathbf{20}$$

$$A \xrightarrow{10} \Delta \xrightarrow{12} E \xrightarrow{\text{σύνολο}} \mathbf{22}$$

$$A \xrightarrow{5} \Gamma \xrightarrow{13} E \xrightarrow{\text{σύνολο}} \mathbf{18}$$

Θέμα 18

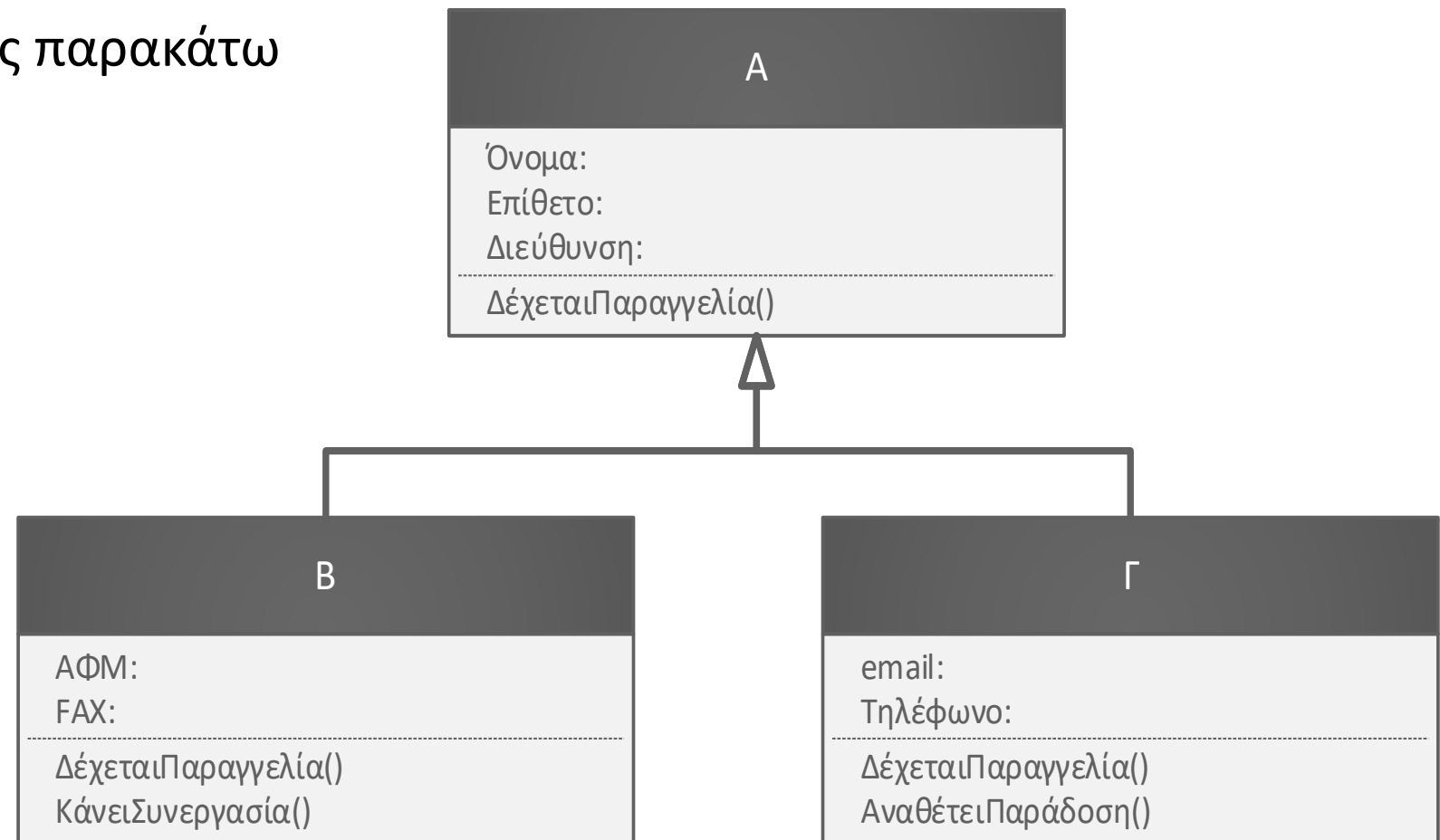
Να οργανώσετε το παρακάτω σύνολο αντικειμένων σε ιεραρχία κληρονομικότητας κλάσεων:
«Πίνακας», «Λίστα», «Τετραγωνικός Πίνακας», «Δυναμικές Δομές», «Δένδρο», «Δομές Δεδομένων»,
«Δυαδικό Δένδρο Αναζήτησης», «Στατικές Δομές», «Δισδιάστατος Πίνακας», «Δυαδικό Δένδρο».



Θέμα 19

Βάσει της διπλανής ιεραρχίας κλάσεων, να χαρακτηρίσετε τις παρακάτω προτάσεις ως Σωστές ή Λανθασμένες:

1. Η κλάση Γ είναι η κλάση πρόγονος (υπερκλάση) και η κλάση Α απόγονός της (υποκλάση). **Λ**
2. Η ιδιότητα «email» είναι ιδιότητα της κλάσης Γ, αλλά όχι της κλάσης Α. **Σ**
3. Ισχύει ότι «ένα Β είναι ένα (is_a) Α». **Σ**
4. Η μέθοδος «ΔέχεταιΠαραγγελία()» είναι πολυμορφική. **Σ**
5. Η ιδιότητα «Επίθετο» είναι μία ιδιότητα της κλάσης Α και της κλάσης Β, αλλά όχι της κλάσης Γ. **Λ**
6. Ένα αντικείμενο της κλάσης Β θα έχει τις ιδιότητες και τις μεθόδους μόνο της κλάσης Β. **Λ**
7. Η μέθοδος «ΚάνειΣυνεργασία()» είναι μία μέθοδος που κληρονομεί η κλάση Β από την κλάση Α. **Λ**
8. Η κλάση Γ έχει συνολικά πέντε ιδιότητες. **Σ**
9. Ένα αντικείμενο της κλάσης Α θα έχει όλες τις ιδιότητες και τις μεθόδους των κλάσεων Β και Γ. **Λ**
10. Η κλάση Β έχει συνολικά 3 μεθόδους. **Λ**



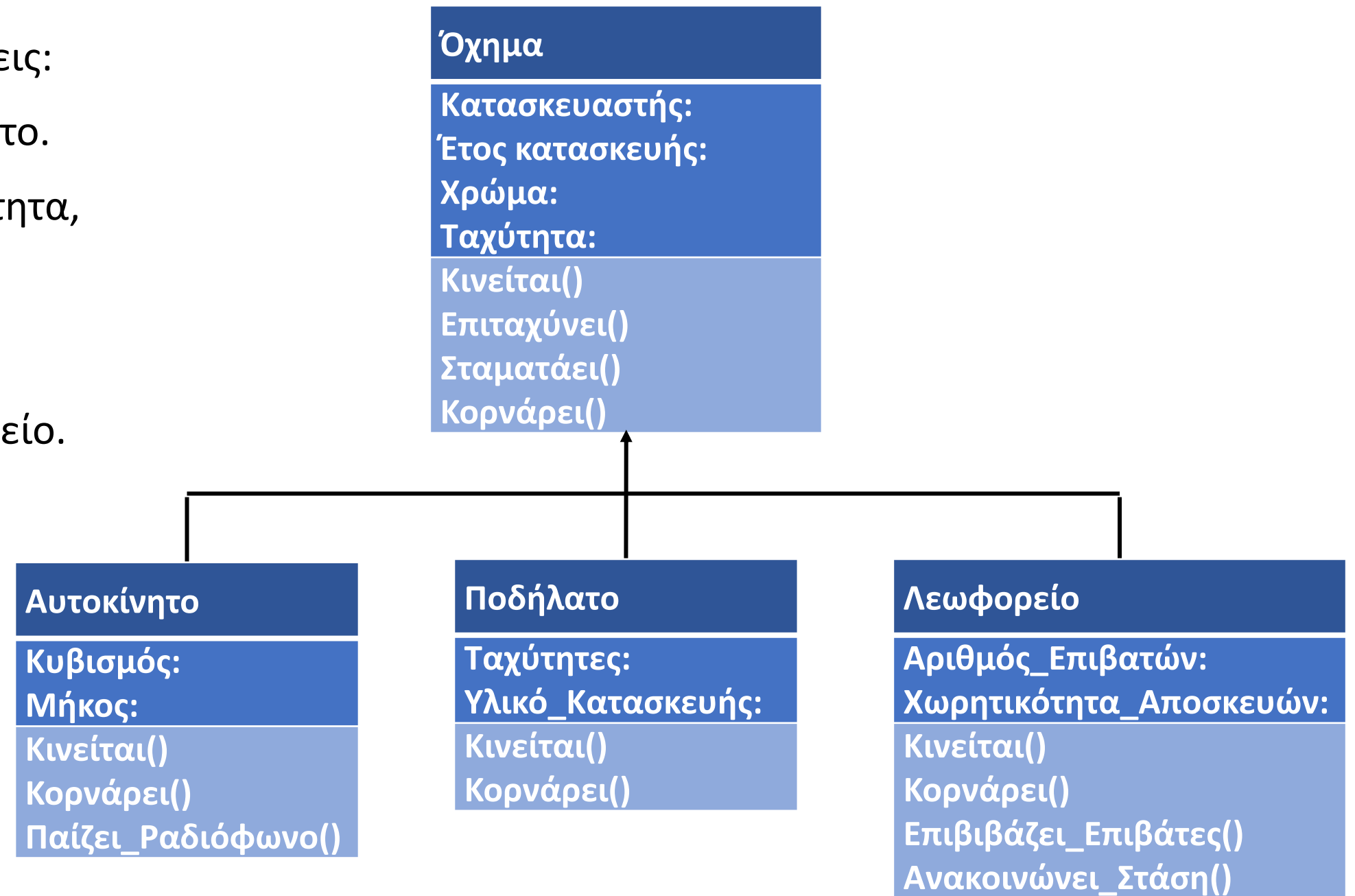
Θέμα 20

Δίνεται το διπλανό διάγραμμα που περιέχει 4 κλάσεις:

- α. Να αναφέρετε τις ιδιότητες της κλάσης Αυτοκίνητο.
Κατασκευαστής, Έτος κατασκευής, Χρώμα, Ταχύτητα,
Κυβισμός, Μήκος.
- β. Να αναφέρετε τις μεθόδους της κλάσης Λεωφορείο.
Κινείται, Επιταχύνει, Σταματάει, Κορνάρει,
Επιβιβάζει_Επιβάτες, Ανακοινώνει_Στάση.
- γ. Να αναφέρετε και να περιγράψετε την ιδιότητα του Αντικειμενοστραφούς Προγραμματισμού που φαίνεται με την επανάληψη της μεθόδου Κινείται() .

Πολυμορφισμός.

Η ιδιότητα του αντικειμενοστραφούς προγραμματισμού με την οποία μια λειτουργία μπορεί να υλοποιείται με πολλούς διαφορετικούς τρόπους.



Θέμα 21

Σε μία σχολή Πληροφορικής σε μεγάλη πόλη της περιφέρειας, ένας φοιτητής, κατά τον πρώτο χρόνο φοίτησης, εξετάζεται γραπτά σε 10 συνολικά μαθήματα. Ο βαθμός που μπορεί να πάρει κάποιος σε μία τέτοια γραπτή δοκιμασία είναι από μηδέν (0) έως και δέκα (10).

Να γραφεί πρόγραμμα σε ΓΛΩΣΣΑ, το οποίο:

Γ1. Να περιλαμβάνει τμήμα δηλώσεων.

Γ2. Για κάθε έναν από τους 70 φοιτητές του τμήματος:

α) να διαβάζει το ονοματεπώνυμό του,

β) να διαβάζει τον βαθμό που πήρε σε κάθε μάθημα, πραγματοποιώντας έλεγχο δεδομένων,

γ) να υπολογίζει και να εμφανίζει το μέσο όρο της βαθμολογίας του.

Γ3. Να εμφανίζει το ποσοστό των φοιτητών που «πέρασαν» (πήραν τουλάχιστον 5) και τα 10 μαθήματα.

Γ4. Να εμφανίζει τα ονόματα των φοιτητών που έχουν τους δύο μεγαλύτερους μέσους όρους, θεωρώντας πως αυτοί είναι διαφορετικοί μεταξύ τους.

Θέμα 21 / Κώδικας

ΠΡΟΓΡΑΜΜΑ Φοιτητές

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: i, C, C5

ΠΡΑΓΜΑΤΙΚΕΣ: ΒΑΘ, S, max1, max2, ΜΟ, ΠΟΣΟΣΤΟ

ΧΑΡΑΚΤΗΡΕΣ: ΟΝ, pos1, pos2

ΑΡΧΗ

C ← 0 ! οι μαθητές που πέρασαν τη βάση

max1 ← -1

max2 ← -2

pos1 ← "

pos2 ← "

ΓΙΑ i **ΑΠΟ** 1 **ΜΕΧΡΙ** 70

ΔΙΑΒΑΣΕ ΟΝ

S ← 0 ! συνολική βαθμολογία κάθε φοιτητή

C5 ← 0 ! τα μαθήματα στα οποία έπιασε τη βάση

ΓΙΑ i **ΑΠΟ** 1 **ΜΕΧΡΙ** 10

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ

ΔΙΑΒΑΣΕ ΒΑΘ

ΜΕΧΡΙΣ_ΟΤΟΥ ΒΑΘ >= 0 **ΚΑΙ** ΒΑΘ <= 10

S ← S + ΒΑΘ

ΑΝ ΒΑΘ >= 5 **ΤΟΤΕ**

C5 ← C5 + 1

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΜΟ ← S / 10

ΓΡΑΨΕ ΜΟ

ΑΝ C5 = 10 **ΤΟΤΕ**

C ← C + 1

ΤΕΛΟΣ_ΑΝ

ΑΝ ΜΟ > max1 **ΤΟΤΕ**

max2 ← max1

pos2 ← pos1

max1 ← ΜΟ

pos1 ← ΟΝ

ΑΛΛΙΩΣ_ΑΝ X > max2 **ΤΟΤΕ**

max2 ← ΜΟ

pos2 ← ΟΝ

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΠΟΣΟΣΤΟ ← C / 70 * 100

ΓΡΑΨΕ 'μαθητές που πέρασαν', ΠΟΣΟΣΤΟ

ΓΡΑΨΕ 'ονόματα των 2 καλύτερων', pos1, pos2

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

Θέμα 22

Μια αποθήκη διανέμει εφόδια σε ομάδες που βρίσκονται σε απομακρυσμένες περιοχές.

Υπάρχει ένα συγκεκριμένο αρχικό απόθεμα εφοδίων.

Κάθε ομάδα ζητάει μια ποσότητα εφοδίων.

Να γραφεί πρόγραμμα σε ΓΛΩΣΣΑ, το οποίο:

Γ1. Να περιλαμβάνει τμήμα δηλώσεων.

Γ2. Δια βάζει το αρχικό διαθέσιμο υπόλοιπο ελέγχοντας πως αυτό είναι θετικός αριθμός.

Γ2. Για κάθε ομάδα διαβάζει τη ζητούμενη ποσότητα θεωρώντας πως δίνεται τιμή μεγαλύτερη του 0.

Η διαδικασία τερματίζεται είτε όταν το υπόλοιπο δεν επαρκεί για τη ζητούμενη ποσότητα είτε όταν εξυπηρετηθούν 100 ομάδες.

Γ3. Υπολογίζει και εμφανίζει το ποσοστό των ομάδων που έλαβαν ποσότητα πάνω από 50 εφόδια.

Γ4. Εμφανίζει το μέγιστο πλήθος εφοδίων που ζητήθηκε.

Γ5. Εμφανίζει πόσες ομάδες ζήτησαν ακριβώς την ίδια ποσότητα με τη μέγιστη.

Θέμα 22 / Κώδικας

ΠΡΟΓΡΑΜΜΑ Εφόδια

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: ΥΠ, Χ, C, C50, Cmax, max

ΠΡΑΓΜΑΤΙΚΕΣ: ΠΟΣΟΣΤΟ

ΑΡΧΗ

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ

ΔΙΑΒΑΣΕ ΥΠ

ΜΕΧΡΙΣ_ΟΤΟΥ ΥΠ > 0

C ← 0

C50 ← 0

max ← -1

ΔΙΑΒΑΣΕ Χ

ΟΣΟ Χ ≤ ΥΠ **ΚΑΙ** C < 100 **ΕΠΑΝΑΛΑΒΕ**

C ← C + 1

ΥΠ ← ΥΠ - Χ

ΑΝ Χ > 50 **ΤΟΤΕ**

C50 ← C50 + 1

ΤΕΛΟΣ_ΑΝ

ΑΝ Χ > max **ΤΟΤΕ**

max ← Χ

Cmax ← 1

ΑΛΛΙΩΣ_ΑΝ Χ = max **ΤΟΤΕ**

Cmax ← Cmax + 1

ΤΕΛΟΣ_ΑΝ

ΑΝ C < 100 **ΤΟΤΕ**

ΔΙΑΒΑΣΕ Χ

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΑΝ C <> 0 **ΤΟΤΕ**

ΠΟΣΟΣΤΟ ← (C50 / C) * 100

ΓΡΑΨΕ ΠΟΣΟΣΤΟ, max, Cmax

ΑΛΛΙΩΣ

ΓΡΑΨΕ "no data"

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

Θέμα 23

Να αναπτύξετε πρόγραμμα σε ΓΛΩΣΣΑ το οποίο υλοποιεί τις λειτουργίες της στοίβας με διαδικασίες και εμφανίζει επαναληπτικά το παρακάτω μενού:

1. Εισαγωγή κιβωτίου (όπου ζητά τον ακέραιο κωδικό του κιβωτίου και τον ωθεί στη στοίβα με τη χρήση της διαδικασίας ΩΘΗΣΗ)
2. Απομάκρυνση κιβωτίου (όπου με τη χρήση της διαδικασίας ΑΠΩΘΗΣΗ απωθεί τον κωδικό από την κορυφή της στοίβας)
3. Εμφάνιση κορυφής στοίβας (όπου απλά εμφανίζει τον κωδικό που βρίσκεται στην κορυφή της στοίβας)
4. Πλήθος κιβωτίων (όπου απλά εμφανίζει το πλήθος των κιβωτίων που βρίσκονται στη στοίβα)
5. Αναζήτηση κιβωτίου *
6. Τερματισμός

Οι επιλογές να εκτελούνται μέχρι ο χρήστης να επιλέξει την επιλογή 6.

* Με την επιλογή 5, καλείται η διαδικασία ΑΝΑΖΗΤΗΣΗ() η οποία κάνει τα εξής:

- Διαβάζει ένα κωδικό κιβωτίου και αναζητά αν αυτός υπάρχει στη στοίβα.
- Η αναζήτηση να πραγματοποιείται χωρίς να αλλοιωθεί τελικά το περιεχόμενο της στοίβας (χρήση βοηθητικής στοίβας).
- Εμφανίζει κατάλληλο μήνυμα αν βρεθεί ή όχι το κιβώτιο.

Να υλοποιήσετε τις λειτουργίες της στοίβας με διαδικασίες.

Θέμα 23 / Κώδικας

ΠΡΟΓΡΑΜΜΑ ΑΠΟΘΗΚΗ

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: S[100], top, ΕΠ, ΚΩΔ, item

ΑΡΧΗ

top ← 0

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ '1. Εισαγωγή κιβωτίου'

ΓΡΑΨΕ '2. Απομάκρυνση κιβωτίου'

ΓΡΑΨΕ '3. Εμφάνιση κορυφής'

ΓΡΑΨΕ '4. Πλήθος κιβωτίων'

ΓΡΑΨΕ '5. Αναζήτηση κιβωτίου'

ΓΡΑΨΕ '6. Τερματισμός'

ΔΙΑΒΑΣΕ ΕΠ

ΕΠΙΛΕΞΕ ΕΠ

ΠΕΡΙΠΤΩΣΗ 1

ΔΙΑΒΑΣΕ ΚΩΔ

ΚΑΛΕΣΕ ΩΘΗΣΗ(S, top, ΚΩΔ)

ΠΕΡΙΠΤΩΣΗ 2

ΚΑΛΕΣΕ ΑΠΩΘΗΣΗ(S, top, ΚΩΔ)

ΠΕΡΙΠΤΩΣΗ 3

ΑΝ top = 0 ΤΟΤΕ

ΓΡΑΨΕ 'Κενή στοίβα'

ΑΛΛΙΩΣ

ΓΡΑΨΕ 'Κορυφή: ', S[top]

ΤΕΛΟΣ_ΑΝ

ΠΕΡΙΠΤΩΣΗ 4

ΓΡΑΨΕ 'Πλήθος κιβωτίων: ', top

ΠΕΡΙΠΤΩΣΗ 5

ΓΡΑΨΕ 'Δώσε κωδικό προς αναζήτηση'

ΔΙΑΒΑΣΕ ΚΩΔ

ΚΑΛΕΣΕ ΑΝΑΖΗΤΗΣΗ(S, top, ΚΩΔ)

ΠΕΡΙΠΤΩΣΗ 6

ΓΡΑΨΕ 'Τερματισμός'

ΠΕΡΙΠΤΩΣΗ ΑΛΛΙΩΣ

ΓΡΑΨΕ 'Λανθασμένη επιλογή'

ΤΕΛΟΣ_ΕΠΙΛΟΓΩΝ

ΜΕΧΡΙΣ_ΟΤΟΥ ΕΠ = 6

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

ΔΙΑΔΙΚΑΣΙΑ ΩΘΗΣΗ(S, top, item)

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: S[100], top, item

ΑΡΧΗ

ΑΝ top = 100 ΤΟΤΕ

ΓΡΑΨΕ 'Υπερχείλιση'

ΑΛΛΙΩΣ

top ← top + 1

S[top] ← item

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΔΙΑΔΙΚΑΣΙΑΣ

!=====

ΔΙΑΔΙΚΑΣΙΑ ΑΠΩΘΗΣΗ(S, top, item)

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: S[100], top, item

ΑΡΧΗ

ΑΝ top = 0 ΤΟΤΕ

ΓΡΑΨΕ 'Υποχείλιση'

ΑΛΛΙΩΣ

item ← S[top]

top ← top - 1

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΔΙΑΔΙΚΑΣΙΑΣ

ΔΙΑΔΙΚΑΣΙΑ ΑΝΑΖΗΤΗΣΗ(S, top, ΚΩΔ)

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: S[100], B[100], top, topB, ΚΩΔ, item

ΛΟΓΙΚΕΣ: done

ΑΡΧΗ

topB ← 0

done ← ΨΕΥΔΗΣ

! Μεταφορά στοιχείων στη βοηθητική στοίβα

ΟΣΟ top > 0 ΚΑΙ done = ΨΕΥΔΗΣ ΕΠΑΝΑΛΑΒΕ

ΚΑΛΕΣΕ ΑΠΩΘΗΣΗ(S, top, item)

ΑΝ item = ΚΩΔ ΤΟΤΕ

done ← ΑΛΗΘΗΣ

ΤΕΛΟΣ_ΑΝ

ΚΑΛΕΣΕ ΩΘΗΣΗ(B, topB, item)

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

! Επαναφορά της αρχικής στοίβας

ΟΣΟ topB > 0 ΕΠΑΝΑΛΑΒΕ

ΚΑΛΕΣΕ ΑΠΩΘΗΣΗ(B, topB, item)

ΚΑΛΕΣΕ ΩΘΗΣΗ(S, top, item)

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΑΝ done = ΑΛΗΘΗΣ ΤΟΤΕ

ΓΡΑΨΕ 'Το κιβώτιο βρέθηκε'

ΑΛΛΙΩΣ

ΓΡΑΨΕ 'Το κιβώτιο δεν βρέθηκε'

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΔΙΑΔΙΚΑΣΙΑΣ

Θέμα 24

Η εταιρεία CITY LOCKER προσφέρει την δυνατότητα στους πελάτες της να παραλαμβάνουν τα δέματα τους από αυτόματα μηχανήματα παραλαβής. Κάθε μηχάνημα διαθέτει 50 θυρίδες.

Να γραφεί πρόγραμμα σε ΓΛΩΣΣΑ το οποίο:

Δ1. Να περιλαμβάνει κατάλληλο τμήμα δηλώσεων.

Δ2. Να διαβάζει στον πίνακα $L[40]$ τους ακέραιους κωδικούς των 40 μηχανημάτων που διατίθενται (χωρίς έλεγχο) και στη συνέχεια, για κάθε ένα από αυτά τα μηχανήματα, να διαβάζει το βάρος σε κιλά του δέματος που βρίσκεται μέσα σε κάθε θυρίδα του μηχανήματος καταχωρίζοντας το στον πίνακα $B[40, 50]$ ελέγχοντας πως δεν δίνεται αρνητικός αριθμός. Εάν δοθεί ως βάρος δέματος το μηδέν, αυτό σημαίνει πως η αντίστοιχη θυρίδα είναι άδεια. Θεωρείστε πως σε κάθε μηχάνημα θα υπάρχει μία τουλάχιστον μη κενή θυρίδα.

Δ3. Θα ζητά από το χρήστη έναν κωδικό και θα καλεί τη συνάρτηση FIND η οποία θα τον αναζητά στον πίνακα L επιστρέφοντας τη αντίστοιχη γραμμή του πίνακα L στην οποία βρέθηκε, ή την τιμή 0 εφόσον δεν βρεθεί.

Στην περίπτωση που ο κωδικός βρεθεί, το πρόγραμμα υπολογίζει και εμφανίζει για το αντίστοιχο μηχάνημα, τον αριθμό της θυρίδας του, στην οποία περιέχεται το βαρύτερο δέμα (μοναδικό).

Σε περίπτωση που ο κωδικός δεν βρεθεί, ρωτά τον χρήστη εάν θα ψάξει για άλλον κωδικό. Εάν αυτός απαντήσει "N" η διαδικασία επαναλαμβάνεται, ενώ τερματίζεται είτε όταν απαντήσει "O" ή όταν ο κωδικός έχει βρεθεί.

Δ4. Θα υπολογίζει και θα εκτυπώνει τους αριθμούς των τριών LOCKER με το μεγαλύτερο μέσο βάρος ανά θυρίδα που χρησιμοποιείται.

Δ5. Θα εμφανίζει τους κωδικούς των LOCKER στα οποία όλα τα δέματα που περιέχονται είναι βαρύτερα από 2 κιλά.

Αν δεν υπάρχει τέτοιο LOCKER να εμφανίζει κατάλληλο μήνυμα.

Δ6. Υλοποιήστε τη συνάρτηση FIND του ερωτήματος Δ3.

Θέμα 24 / Κώδικας

ΠΡΟΓΡΑΜΜΑ LOCKERS

!Δ1

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: i, j, L[40], key, pos, C, temp2

ΠΡΑΓΜΑΤΙΚΕΣ: B[40, 50], max, posj, S, MO[40], temp1

ΧΑΡΑΚΤΗΡΕΣ: ΑΠ

ΛΟΓΙΚΕΣ: done

ΑΡΧΗ

!Δ2

ΑΠ ← 'N'

ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 40

ΔΙΑΒΑΣΕ L[i]

ΓΙΑ j ΑΠΟ 1 ΜΕΧΡΙ 50

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ

ΔΙΑΒΑΣΕ B[i, j]

ΜΕΧΡΙΣ_ΟΤΟΥ B[i, j] >= 0

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

!Δ3

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ

ΔΙΑΒΑΣΕ key

pos ← FIND(L, key)

ΑΝ pos <> 0 ΤΟΤΕ

max ← -1

ΓΙΑ j ΑΠΟ 1 ΜΕΧΡΙ 50

ΑΝ B[pos, j] > max ΤΟΤΕ

max ← B[pos, j]

posj ← j

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ posj

ΑΛΛΙΩΣ

ΓΡΑΨΕ 'Θα ψάξω για άλλο; (N/O)'

ΔΙΑΒΑΣΕ ΑΠ

ΤΕΛΟΣ_ΑΝ

ΜΕΧΡΙΣ_ΟΤΟΥ pos <> 0 Η ΑΠ = 'O'

!Δ4

ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 40

C ← 0

S ← 0

ΓΙΑ j ΑΠΟ 1 ΜΕΧΡΙ 50

ΑΝ B[i, j] <> 0 ΤΟΤΕ

S ← S + B[i, j]

C ← C + 1

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΜΟ[i] ← S / C

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΙΑ i ΑΠΟ 2 ΜΕΧΡΙ 40

ΓΙΑ j ΑΠΟ 40 ΜΕΧΡΙ i ΜΕ_ΒΗΜΑ -1

ΑΝ ΜΟ[j - 1] < ΜΟ[j] ΤΟΤΕ

temp1 ← ΜΟ[j - 1]

ΜΟ[j - 1] ← ΜΟ[j]

ΜΟ[j] ← temp1

temp2 ← L[j - 1]

L[j - 1] ← L[j]

L[j] ← temp2

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ L[1], L[2], L[3]

!Δ5

C ← 0

ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 40

done ← ΑΛΗΘΗΣ

ΓΙΑ j ΑΠΟ 1 ΜΕΧΡΙ 50

ΑΝ B[i, j] <= 2 ΚΑΙ B[i, j] <> 0 ΤΟΤΕ

done ← ΨΕΥΔΗΣ

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΑΝ done = ΑΛΗΘΗΣ ΤΟΤΕ

ΓΡΑΨΕ L[i]

C ← C + 1

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΑΝ C = 0 ΤΟΤΕ

ΓΡΑΨΕ 'no data'

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

!Δ6 =====

ΣΥΝΑΡΤΗΣΗ FIND(L, key): ΑΚΕΡΑΙΑ

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: i, L[40], key, pos

ΛΟΓΙΚΕΣ: done

ΑΡΧΗ

i ← 1

pos ← 0

done ← ΨΕΥΔΗΣ

ΟΣΟ done = ΨΕΥΔΗΣ ΚΑΙ i <= 40 ΕΠΑΝΑΛΑΒΕ

ΑΝ L[i] = key ΤΟΤΕ

pos ← i

done ← ΑΛΗΘΗΣ

ΤΕΛΟΣ_ΑΝ

i ← i + 1

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

FIND ← pos

ΤΕΛΟΣ_ΣΥΝΑΡΤΗΣΗΣ

**Εύχομαι ο καθένας σας να φτάσει ή
και να ξεπεράσει τις προσδοκίες του.**

Χάρηκα που σας είχα.

Ελπίζω να χαρήκατε κι εσείς μαζί μου.

Φιλιά σε όλους.