

**Επαναληπτικό Διαγώνισμα για προπονημέν(ες-ους)  
στην "Πληροφορική" της Γ'τάξης του Γενικού Λυκείου  
Σχολικό έτος 2024-2025**

**ΘΕΜΑ Α**

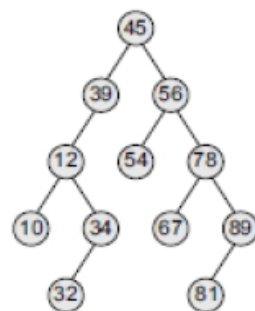
**A1.** Να γράψετε στο τετράδιό σας τον αριθμό καθεμιάς από τις παρακάτω προτάσεις 1-5 και δίπλα τη λέξη **ΣΩΣΤΟ**, αν είναι σωστή, ή τη λέξη **ΛΑΘΟΣ** αν είναι λανθασμένη.

1. Η σωστή και πλήρης αποσαφήνιση των δεδομένων και των ζητούμενων ενός προβλήματος επιτυγχάνεται κατά το στάδιο της ανάλυσής του.
2. Κατά την διαδοχική κλήση πολλών υποπρογραμμάτων, η αποθήκευση και η ανάκτηση των διευθύνσεων επιστροφής από τον μεταφραστή, γίνεται με τη μέθοδο FIFO.
3. Τα δυαδικά δένδρα αναζήτησης συνδυάζουν τα πλεονεκτήματα των δυναμικών δομών δεδομένων, δηλαδή τη δυνατότητα αυξομείωσης του πλήθους των κόμβων τους κατά την εκτέλεση του προγράμματος, αλλά και τα πλεονεκτήματα των ταξινομημένων πινάκων όσον αφορά την πράξη της αναζήτησης.
4. Η ιδιότητα του πολυμορφισμού του αντικειμενοστραφούς προγραμματισμού, δίνει τη δυνατότητα στις κλάσεις απογόνους να τροποποιούν τις ιδιότητες που κληρονομούν από τις κλάσεις προγόνους, αλλά όχι και τις μεθόδους τους.
5. Ένα εκτελέσιμο πρόγραμμα τελείως ανεξάρτητο από το αρχικό (πηγαίο) παράγεται αποκλειστικά με τη χρήση διερμηνευτή.

**Μονάδες 10**

**A2.** Δίνεται το δυαδικό δένδρο αναζήτησης:

Να σχεδιάσετε την τελική μορφή του δυαδικού δένδρου αναζήτησης, μετά την εισαγωγή 5 νέων κόμβων με τιμές 47, 79, 40, 70, 36, οι οποίοι να αποτελούν φύλλα του νέου δένδρου.



**Μονάδες 5**

**A3.** Ο παρακάτω αλγόριθμος με τις εντολές του αριθμημένες, αποτελεί τμήμα μη δομημένου προγράμματος. Ακολουθεί μόνο τον κανόνα της δομής της ακολουθίας (εκτέλεση εντολών κατ' αύξουσα σειρά του αριθμού τους), ενώ τις δομές επιλογής και επανάληψης αντικαθιστά η εντολή άλματος GOTO. Να γράψετε αλγόριθμο σχεδιασμένο με τις αρχές του δομημένου προγραμματισμού, που να εκτελεί τις ίδιες λειτουργίες. Δηλαδή να αντικαταστήσετε τις εντολές GOTO με τις

κατάλληλες δομές επιλογής και επανάληψης, και να αφαιρέσετε την αρίθμηση των εντολών.

```
10 Αρχή
20 Μονοί ← 0
30 Ζυγοί ← 0
40 Διάβασε α
50 Αν α=0 τότε GOTO 120
60 Αν α mod 2=0 τότε GOTO 90
70 Μονοί ← Μονοί + 1
80 GOTO 100
90 Ζυγοί ← Ζυγοί + 1
100 Διάβασε α
110 GOTO 50
120 Εμφάνισε "Μονοί=",Μονοί,"Ζυγοί=",Ζυγοί
130 Τέλος
```

**Μονάδες 8**

**A4.** Στην πρώτη στήλη του επόμενου πίνακα αναγράφονται τρεις **Τεχνικές σχεδίασης προγραμμάτων** και στη δεύτερη στήλη κάποια πιθανά **Βασικά χαρακτηριστικά τους ή Πλεονεκτήματα**. Να γράψετε δίπλα σε κάθε γράμμα της πρώτης στήλης τον αριθμό ή τους αριθμούς της δεύτερης στήλης που θεωρείτε ότι αντιστοιχίζεται σωστά.

| Τεχνικές σχεδίασης προγραμμάτων       | Βασικά χαρακτηριστικά - Πλεονεκτήματα                                                                                          |
|---------------------------------------|--------------------------------------------------------------------------------------------------------------------------------|
| A. Δομημένος προγραμματισμός          | 1. Φυσικός τρόπος επίλυσης προβλημάτων                                                                                         |
| B. Τμηματικός προγραμματισμός         | 2. Στηρίζεται στη χρήση τριών και μόνο στοιχειωδών λογικών δομών, τη δομή ακολουθίας, τη δομή επιλογής και τη δομή επανάληψης. |
| Γ. Αντικειμενοστραφής προγραμματισμός | 3. Επεκτείνει τις δυνατότητες των γλωσσών προγραμματισμού                                                                      |
|                                       | 4. Πρωτεύοντα δομικά στοιχεία είναι τα δεδομένα και όχι οι ενέργειες                                                           |

**Μονάδες 2**

## **ΘΕΜΑ Β**

**B1.** Ζητήθηκε να σχεδιασθεί αλγόριθμος, ο οποίος να διαβάζει τα ονόματα των τριών Ολυμπιονικών σ' ένα αγώνισμα στίβου, στη συνέχεια να διαβάζει τα ονόματα του χάλκινου και αργυρού νικητή, και τελικά να εμφανίζει το όνομα του χρυσού νικητή.

Ως απάντηση προτάθηκαν οι 4 αλγόριθμοι που ακολουθούν:

|                                                                                                                                                                                                                                                                                                                                                                                                                                |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p>Αλγόριθμος1<br/> Διάβασε on1,on2,on3<br/> Διάβασε επ1,επ2<br/> Αν επ1=on1 και επ2=on2 τότε<br/> Εμφάνισε on3<br/> αλλιώς_αν επ1=on1 και επ2=on3 τότε<br/> Εμφάνισε on2<br/> αλλιώς<br/> Εμφάνισε on1<br/> Τέλος_αν<br/> Τέλος Αλγόριθμος</p>                                                                                                                                                                                | <p>Αλγόριθμος2<br/> Διάβασε on1,on2,on3<br/> Διάβασε επ1,επ2<br/> Αν επ1=on1 και επ2=on2 τότε<br/> Εμφάνισε on3<br/> Τέλος_αν<br/> Αν επ1=on1 και επ2=on3 τότε<br/> Εμφάνισε on2<br/> Τέλος_αν<br/> Αν επ1=on2 και επ2=on3 τότε<br/> Εμφάνισε on1<br/> Τέλος_αν<br/> Τέλος Αλγόριθμος2</p>                                                                                                                                                                                                    |
| <p>Αλγόριθμος3<br/> Διάβασε on1,on2,on3<br/> Διάβασε επ1,επ2<br/> Αν επ1=on1 και επ2=on2 τότε<br/> Εμφάνισε on3<br/> αλλιώς_αν επ1=on1 και επ2=on3 τότε<br/> Εμφάνισε on2<br/> αλλιώς_αν επ1=on2 και επ2=on1 τότε<br/> Εμφάνισε on3<br/> αλλιώς_αν επ1=on2 και επ2=on3 τότε<br/> Εμφάνισε on1<br/> αλλιώς_αν επ1=on3 και επ2=on1 τότε<br/> Εμφάνισε on2<br/> αλλιώς<br/> Εμφάνισε on1<br/> Τέλος_αν<br/> Τέλος Αλγόριθμος3</p> | <p>Αλγόριθμος4<br/> Διάβασε on1,on2,on3<br/> Διάβασε επ1,επ2<br/> Αν επ1=on1 και επ2=on2 τότε<br/> Εμφάνισε on3<br/> Τέλος_αν<br/> Αν επ1=on1 και επ2=on3 τότε<br/> Εμφάνισε on2<br/> Τέλος_αν<br/> Αν επ1=on2 και επ2=on1 τότε<br/> Εμφάνισε on3<br/> Τέλος_αν<br/> Αν επ1=on2 και επ2=on3 τότε<br/> Εμφάνισε on1<br/> Τέλος_αν<br/> Αν επ1=on3 και επ2=on1 τότε<br/> Εμφάνισε on2<br/> Τέλος_αν<br/> Αν επ1=on3 και επ2=on2 τότε<br/> Εμφάνισε on1<br/> Τέλος_αν<br/> Τέλος Αλγόριθμος4</p> |

α) Να ελέγξετε την ορθότητά τους. Σε ποια κατηγορία ανήκει το σφάλμα κάθε λανθασμένου αλγορίθμου;

**Μονάδες 2**

β) Για καθένα απ' τους αλγορίθμους που λειτουργούν σωστά, να μετρήσετε αθροιστικά το πλήθος των συγκριτικών και λογικών πράξεων που χρειάζεται να εκτελεστούν:

**I)** όταν εκτελείται η πρώτη εντολή **Εμφάνισε** του Αλγορίθμου (που αποτελεί την καλύτερη περίπτωση με το μικρότερο υπολογιστικό κόστος, δηλαδή με το ελάχιστο δυνατό πλήθος πράξεων).

**II)** όταν εκτελείται η τελευταία εντολή **Εμφάνισε** του Αλγορίθμου (που αποτελεί την χειρότερη περίπτωση με το μεγαλύτερο υπολογιστικό κόστος, δηλαδή με το μέγιστο δυνατό πλήθος πράξεων).

**Μονάδες 2**

γ) Αν θεωρήσουμε ως αποδοτικότερο αλγόριθμο αυτόν που κάνει τις λιγότερες πράξεις (συγκριτικές και λογικές συνολικά) στη χειρότερη περίπτωση του, απ' τις αντίστοιχες πράξεις των άλλων αλγορίθμων, να βρείτε ποιος απ' τους ορθούς αλγορίθμους είναι ο πιο αποδοτικός.

**Μονάδα 1**

δ) Να σχεδιάσετε για το ίδιο πρόβλημα έναν αποδοτικότερο αλγόριθμο που δεν κάνει λογικές πράξεις, δηλαδή δεν κάνει χρήση λογικών τελεστών. Πόσες συγκρίσεις απαιτούνται για την καλύτερη και τη χειρότερη περίπτωση του;

**Μονάδες 5**

**B2. α)** Ποια ή ποιες απ' τις 4 επόμενες συναρτήσεις υπολογίζει σωστά το

$$\text{άθροισμα : } 1 + 11 + 111 + 1111 + \dots + \overbrace{111\dots 111}^{n\text{-ψηφία}}$$

**ΣΥΝ1 (ν) : ΑΚΕΡΑΙΑ**

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: ν, α, Σ, π, ι

ΑΡΧΗ

α ← 1

Σ ← 1

π ← 1

ΓΙΑ ι ΑΠΟ 2 ΜΕΧΡΙ ν

π ← π\*10

α ← α+π

Σ ← Σ+α

ΤΕΛΟΣ\_ΕΠΑΝΑΛΗΨΗΣ

ΣΥΝ1 ← Σ

ΤΕΛΟΣ\_ΣΥΝΑΡΤΗΣΗΣ

**ΣΥΝ3 (ν) : ΑΚΕΡΑΙΑ**

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: ν, α, Σ, ι

ΑΡΧΗ

α ← 1

Σ ← 1

ΓΙΑ ι ΑΠΟ 2 ΜΕΧΡΙ ν

α ← α\*11

Σ ← Σ+α

ΤΕΛΟΣ\_ΕΠΑΝΑΛΗΨΗΣ

ΣΥΝ3 ← Σ

ΤΕΛΟΣ\_ΣΥΝΑΡΤΗΣΗΣ

**ΣΥΝ2 (ν) : ΑΚΕΡΑΙΑ**

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: ν, α, Σ, ι

ΑΡΧΗ

α ← 1

Σ ← 1

ΓΙΑ ι ΑΠΟ 2 ΜΕΧΡΙ ν

α ← α+10<sup>(ι-1)</sup>

Σ ← Σ+α

ΤΕΛΟΣ\_ΕΠΑΝΑΛΗΨΗΣ

ΣΥΝ2 ← Σ

ΤΕΛΟΣ\_ΣΥΝΑΡΤΗΣΗΣ

**ΣΥΝ4 (ν) : ΑΚΕΡΑΙΑ**

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: ν, α, Σ, ι

ΑΡΧΗ

α ← 1

Σ ← 1

ΓΙΑ ι ΑΠΟ 2 ΜΕΧΡΙ ν

α ← α\*10+1

Σ ← Σ+α

ΤΕΛΟΣ\_ΕΠΑΝΑΛΗΨΗΣ

ΣΥΝ4 ← Σ

ΤΕΛΟΣ\_ΣΥΝΑΡΤΗΣΗΣ

**Μονάδες 6**

β) **I)** Να γράψετε συνάρτηση **ΣΥΝ5 (ψ, ν)** η οποία παραμετροποιεί το προηγούμενο πρόβλημα. Να έχει ένα επιπλέον όρισμα ψ ώστε να δέχεται επιπλέον ένα μονοψήφιο θετικό ακέραιο στο διάστημα [1, 9] και να υπολογίζει και να

επιστρέφει το άθροισμα:  $\psi + \overbrace{\psi\psi + \psi\psi\psi + \psi\psi\psi\psi + \dots}^{n\text{-ψηφία}} + \psi\psi\psi\dots\psi\psi\psi$

π.χ. για ψ=6, το άθροισμα:  $6 + \overbrace{66 + 666 + 6666 + \dots}^{n\text{-ψηφία}} + 666\dots 666$

**Μονάδες 3**

**II)** Να γράψετε κύριο πρόγραμμα το οποίο να διαβάζει τις τιμές των μεταβλητών  $\psi$  και  $\nu$ , να κάνει έλεγχο εγκυρότητας για την τιμή της  $\psi$ , να καλεί την συνάρτηση ΣΥΝ5 και τέλος να εμφανίζει το ζητούμενο άθροισμα.

**Μονάδες 6**

### **ΘΕΜΑ Γ**

Να γράψετε πρόγραμμα σε "ΓΛΩΣΣΑ" το οποίο:

**Γ1.** Να περιλαμβάνει τμήμα δηλώσεων.

**Μονάδα 1**

**Γ2.** Να διαβάζει έναν θετικό ακέραιο αριθμό με τον κατάλληλο έλεγχο σωστής εισαγωγής. (Το μέγεθος του μπορεί να είναι τόσο μεγάλο όσο επιτρέπει η μνήμη και η αριθμητική του υπολογιστικού συστήματος).

**Μονάδες 2**

**Γ3.** Να εμφανίζει τη μέγιστη συνεχόμενη αναγραφή του ίδιου ψηφίου στον αριθμό, καθώς και το ψηφίο που επαναλαμβάνεται. Σε περίπτωση ίσης μέγιστης συνεχόμενης αναγραφής διαφορετικών ψηφίων στον αριθμό, να εμφανίζεται το ψηφίο με τη μεγαλύτερη τιμή.

**Μονάδες 7**

**Γ4.** Να εμφανίζει το πλήθος των ψηφίων του αριθμού.

**Μονάδες 2**

**Γ5.** Να εμφανίζει την Παλινδρομική (καρκινική) μορφή του αριθμού (αναγραφή των ψηφίων του με αντίστροφη σειρά).

**Μονάδες 6**

**Γ6.** Να εμφανίζει ποιο ή ποια ψηφία εμφανίζονται ανεξαρτήτως σειράς τις περισσότερες φορές στον αριθμό.

**Μονάδες 7**

π.χ. για τους αριθμούς:

**776669577**

μέγιστη διαδοχή: 3 φορές το 6, αριθμός με 9 ψηφία, παλίνδρομος: 775966677  
μέγιστη εμφάνιση: 4 φορές το 7.

**5556099900**

μέγιστη διαδοχή: 3 φορές το 9, αριθμός με 10 ψηφία, παλίνδρομος: 99906555  
μέγιστη εμφάνιση: 3 φορές το 0, το 5, το 9.

**594**

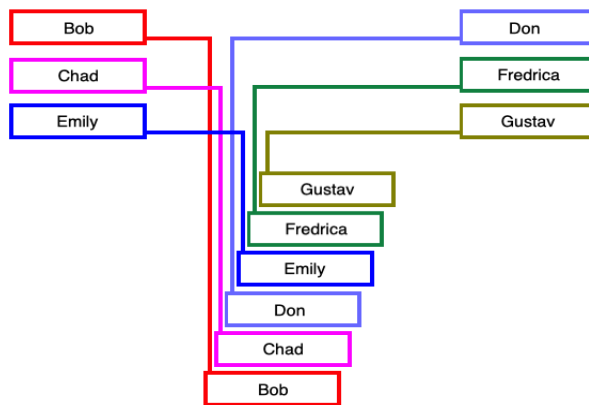
μέγιστη διαδοχή: 1 φορά το 9, αριθμός με 3 ψηφία, παλίνδρομος: 495  
μέγιστη εμφάνιση: 1 φορά το 4, το 5, το 9.

**7**

μέγιστη διαδοχή: 1 φορά το 7, αριθμός με 1 ψηφίο, παλίνδρομος: 7  
μέγιστη εμφάνιση: 1 φορά το 7.

## ΘΕΜΑ Δ

Μία απ' τις βασικές λειτουργίες επί των δομών δεδομένων είναι και η συγχώνευση, κατά την οποία δύο ή περισσότερες δομές συνενώνονται σε μία ενιαία δομή. Ζητείται να συγχωνεύσετε δύο ταξινομημένες στοίβες ΣΤΒ1, ΣΤΒ2 σε μία τρίτη ταξινομημένη στοίβα ΣΤΒ3. Τονίζεται ότι οι μόνες επιτρεπτές λειτουργίες-πράξεις στις στοίβες είναι η **Ωθηση** και η **Απόθεση**, οι οποίες να υλοποιηθούν ως υποπρογράμματα (ερωτήματα Δ5 I και II) και να καλούνται όπου απαιτείται στα παρακάτω ερωτήματα.



Πιο συγκεκριμένα, ζητείται να γράψετε πρόγραμμα σε "Γλώσσα", το οποίο:

**Δ1.** Να περιλαμβάνει κατάλληλο τμήμα δηλώσεων.

**Μονάδα 1**

**Δ2. Να διαβάσει** δύο αλφαβητικούς καταλόγους ονομάτων που περιέχουν το πολύ 100 ονόματα ο καθένας και να καταχωρίζει τον καθένα απ' αυτούς σε μία ξεχωριστή στοίβα. Η υλοποίηση τους να γίνεται στατικά με τη βοήθεια των πινάκων ΣΤΒ1[200] και ΣΤΒ2[200]. Διευκρινίζεται ότι η δέσμευση 100 επιπλέον θέσεων σε κάθε πίνακα γίνεται για να υπάρχει συμβατότητα (ίδιο μέγεθος) με τον τελικό συγχωνευμένο πίνακα ΣΤΒ3[200], ο οποίος θα περιέχει τα ονόματα και των δύο καταλόγων και επομένως απαιτεί 200 θέσεις. (Η συμβατότητα απαιτείται για τη δυνατότητα κλήσης των υποπρογραμμάτων των πράξεων Ωθησης και Απόθεσης για κάθε μία απ' τις τρεις στοίβες ΣΤΒ1, ΣΤΒ2, ΣΤΒ3 που σημαίνει ότι πρέπει οι τρεις πίνακες να είναι ίδιου μεγέθους). Εν ολίγοις οι πίνακες ΣΤΒ1 και ΣΤΒ2 θα περιέχουν αρχικά το πολύ 100 ονόματα ο καθένας, στις 100 πρώτες θέσεις του. Στις υπόλοιπες 100 δεν θα καταχωριστούν τιμές.

Τα ονόματα να ωθούνται σε κάθε στοίβα κατ' αλφαβητική σειρά (να γίνεται κατάλληλος έλεγχος πριν από κάθε ώθηση και σε περίπτωση παραβίασης να ξαναδιαβάζεται νέο όνομα μέχρι αυτό να μη παραβιάζει την αλφαβητική ταξινόμηση). Η εισαγωγή κάθε καταλόγου να τερματίζεται όταν δοθεί ως όνομα το αλφαριθμητικό 'ΩΩ' ή όταν συμπληρωθούν 100 ονόματα.

Θεωρείστε ότι κάθε κατάλογος περιέχει τουλάχιστον ένα όνομα και ότι μπορεί να υπάρχουν συνωνυμίες.

Επειδή η όλη διαδικασία του ερωτήματος Δ2 θα πρέπει να επαναληφθεί για τη δημιουργία κάθε μίας απ' τις δύο στοίβες, είναι προτιμότερο να υλοποιηθεί με κατάλληλο υποπρόγραμμα (ερώτημα Δ5 III) το οποίο να καλείται για τη δημιουργία κάθε στοίβας ξεχωριστά.

**Μονάδα 1**

**Δ3.** Να συγχωνεύει τις στοίβες ΣΤΒ1 και ΣΤΒ2 σε μία τρίτη ΣΤΒ3. Αυτή η πράξη υλοποιείται γενικά με τα επόμενα 8 βήματα:

**Βήμα1:** Να απωθεί το πρώτο όνομα από καθεμιά απ' τις στοίβες ΣΤΒ1, ΣΤΒ2.

**Βήμα2:** Να ελέγχει αν οι στοίβες ΣΤΒ1, ΣΤΒ2 έχουν από ένα μόνο όνομα και στην περίπτωση αυτή να δημιουργεί τη ΣΤΒ3 και να πηγαίνει στο **Βήμα8**.

**Βήμα3:** Να ελέγχει αν καμία απ' τις δύο στοίβες ΣΤΒ1, ΣΤΒ2 δεν είναι άδεια. Σε περίπτωση που αυτό δεν αληθεύει να πηγαίνει στο **Βήμα6**.

**Βήμα4:** Να συγκρίνει τα δύο ονόματα και το αλφαβητικά μεγαλύτερο ή ίσο, να το ωθεί στη ΣΤΒ3 και να απωθεί το επόμενο όνομα της στοίβας απ' την οποία απωθήθηκε το όνομα που ωθήθηκε στην ΣΤΒ3.

**Βήμα5:** Επιστροφή στο Βήμα3 και επανάληψη των βημάτων 3 έως 5.

Στα επόμενα βήματα πρέπει να ωθηθούν στη στοίβα ΣΤΒ3 τα υπόλοιπα ονόματα απ' τις στοίβες ΣΤΒ1, ΣΤΒ2 που δεν ωθήθηκαν με την προηγούμενη επαναληπτική διαδικασία. Εδώ πρέπει να δοθεί προσοχή στα εξής: Μετά τον τερματισμό της προηγούμενης επαναληπτικής διαδικασίας, το τελευταίο όνομα που απωθήθηκε απ' τη στοίβα που άδειαζε, δεν έχει ωθηθεί στην ΣΤΒ3 διότι με την τελευταία απώθηση ο δείκτης TOP γίνεται 0, δηλαδή η στοίβα είναι άδεια ενώ το όνομα που απωθήθηκε δεν έχει συγκριθεί με το όνομα που έχει απωθηθεί τελευταίο από τη στοίβα που δεν έχει αδειάσει. Επομένως εκκρεμεί να ωθηθεί στη στοίβα ΣΤΒ3, στη σωστή όμως αλφαβητικά θέση, μαζί με τα υπόλοιπα ονόματα της στοίβας που δεν έχει αδειάσει. Ακόμη πρέπει να τονιστεί ότι και το όνομα, που απωθήθηκε τελευταίο απ' τη στοίβα που δεν έχει αδειάσει, δεν έχει κι αυτό ωθηθεί στη στοίβα ΣΤΒ3.

**Βήμα6:** Να ωθεί στη στοίβα ΣΤΒ3 το μεγαλύτερο αλφαβητικά όνομα απ' τα δύο ονόματα που προαναφέρθηκαν και εκκρεμούν να ωθηθούν και να διαπιστώνει ποια στοίβα περιέχει το μικρότερο αλφαβητικά όνομα το οποίο εκκρεμεί να ωθηθεί στο επόμενο βήμα.

**Βήμα7:** Ανάλογα με ποια στοίβα άδειαζε και σε ποια στοίβα ανήκει το όνομα που εκκρεμεί να ωθηθεί απ' το Βήμα6, να κάνει τις κατάλληλες κλήσεις των υποπρογραμμάτων Δ5 IV και V ώστε να γίνουν οι συμπληρωματικές ωθήσεις όλων των υπόλοιπων ονομάτων στη ΣΤΒ3.

**Βήμα8:** Τέλος

**Μονάδες 7**

**Δ4.** Να εκτυπώνει με αλφαβητική σειρά τα ονόματα της στοίβας ΣΤΒ3. **Μονάδα 1**

**Δ5.** Να συντάξετε τα επόμενα υποπρογράμματα:

**I.** **Ωθηση (Όνομα, top, Στοίβα)** που ωθεί την αλφαριθμητική τιμή της παραμέτρου **Όνομα** στη θέση **top+1** του πίνακα **Στοίβα** (η κλασική λειτουργία-πράξη της στοίβας) **Μονάδες 2**

**II.** **Απώθηση (Όνομα, top, Στοίβα)** που απωθεί την αλφαριθμητική τιμή από τη θέση **top** του πίνακα **Στοίβα** και την επιστρέφει μέσω της παραμέτρου **Όνομα** (η κλασική λειτουργία-πράξη της στοίβας) **Μονάδες 2**

**III.** **Δημιουργία\_στοίβας (top, Στοίβα)** που διαβάζει ονόματα με αλφαβητική σειρά και δημιουργεί την ταξινομημένη **Στοίβα** ονομάτων όπως περιγράφεται στο ερώτημα Δ2. **Μονάδες 3**

**IV.** **Συμπλήρωση\_1 (top, Στοίβα, top3, Στοίβα3)** που απωθεί διαδοχικά ένα-ένα όλα τα ονόματα της στοίβας **Στοίβα** απ' τη θέση **top** μέχρι να αδειάσει και τα ωθεί ένα-ένα στην στοίβα **Στοίβα3** αρχίζοντας απ' τη θέση **top3+1**.

Το υποπρόγραμμα αυτό να καλείται για να ωθήσει στη ΣΤΒ3 τα υπόλοιπα ονόματα που περιέχει η στοίβα που δεν έχει αδειάσει στα βήματα 3-5 του ερωτήματος Δ3. **Μονάδες 2**

**V.** **Συμπλήρωση\_2 (Όνομα, top, Στοίβα, top3, Στοίβα3)** που απωθεί διαδοχικά ένα-ένα όλα τα ονόματα της στοίβας **Στοίβα** απ' τη θέση **top** και τα ωθεί ένα-ένα στην στοίβα **Στοίβα3** απ' τη θέση **top3+1** και επάνω, φροντίζοντας συγχρόνως να ωθηθεί η αλφαριθμητική τιμή που περιέχει η παράμετρος **Όνομα** στην κατάλληλη θέση της στοίβας **Στοίβα3** ώστε να διατηρηθεί η αλφαβητική σειρά των ονομάτων στη Στοίβα3.

Το υποπρόγραμμα αυτό να καλείται για να ωθήσει στη ΣΤΒ3, τα υπόλοιπα ονόματα που περιέχει η στοίβα που δεν έχει αδειάσει στα βήματα 3-5 του ερωτήματος Δ3, και συγχρόνως να ωθήσει, στην σωστή αλφαβητικά θέση της στοίβας ΣΤΒ3, το όνομα που εκκρεμεί να ωθηθεί (στο Βήμα6), στην περίπτωση που αυτό ανήκει στη στοίβα που άδειαζε στα βήματα 3-5 του ερωτήματος Δ3. **Μονάδες 6**