

ΦΡΟΝΤΙΣΤΗΡΙΟ «ΕΝΑ»

ΑΝΑΠΤΥΞΗ ΕΦΑΡΜΟΓΩΝ ΣΕ ΠΡΟΓΡΑΜΜΑΤΙΣΤΙΚΟ ΠΕΡΙΒΑΛΛΟΝ

ΤΕΧΝΟΛΟΓΙΚΗΣ ΚΑΤΕΥΘΥΝΣΗΣ Γ' ΛΥΚΕΙΟΥ

ΘΕΜΑ 1°

A. Να γράψετε στο τετράδιο σας τον αριθμό καθεμιάς από τις παρακάτω προτάσεις και δίπλα τη λέξη Σωστό ή τη λέξη Λάθος αν είναι σωστή ή λανθασμένη αντίστοιχα

1. Η γλώσσα προγραμματισμού PASCAL είναι γλώσσα υψηλού επιπέδου κατάλληλη για τη δημιουργία δομημένων προγραμμάτων
2. Οι διαστάσεις ενός πίνακα μπορούν να αλλάξουν κατά την διάρκεια εκτέλεσης ενός αλγορίθμου
3. Η μέθοδος «Διαίρει και Βασίλευε» αποτελεί μια από τις τυποποιημένες κατηγορίες τεχνικών που χρησιμοποιούνται για την επίλυση συγκεκριμένων προβλημάτων
4. Για τον πολλαπλασιασμό ενός δυαδικού αριθμού με το δύο μπορούμε να μεταφέρουμε όλα τα ψηφία του αριθμού κατά μία θέση προς τα δεξιά και να προσθέσουμε ένα μηδέν στο τέλος
5. Η χρησιμοποίηση πινάκων στον προγραμματισμό προσφέρει το πλεονέκτημα της ελάττωσης της συνολικής μνήμης που θα απαιτούσε το πρόγραμμα χωρίς την χρήση πινάκων

B. Δίνεται το παρακάτω τμήμα αλγορίθμου σε φυσική γλώσσα με βήματα. Να γραφεί το ισοδύναμο του με κωδικοποίηση.

- Βήμα 1: Θέσε στην μεταβλητή «S» την τιμή μηδέν
Βήμα 2: Θέσε στην μεταβλητή «i» την τιμή ένα
Βήμα 3: Αν το «i» είναι μικρότερο ή ίσο με δέκα τότε πήγαινε στο Βήμα 4 αλλιώς πήγαινε στο βήμα 7
Βήμα 4: Αύξησε την μεταβλητή «S» κατά 5
Βήμα 5: Αύξησε την μεταβλητή «i» κατά 1
Βήμα 6: Πήγαινε στο βήμα 2
Βήμα 7: Εμφάνισε τα αποτελέσματα

Γ. Να γράψετε ποια αλγοριθμικά κριτήρια δεν ικανοποιεί το παρακάτω τμήμα αλγορίθμου και γιατί

Διάβασε α,β

Αν $\beta < 3$ τότε

$Z \leftarrow \alpha / (\beta - 3)$

αλλιώς

$Z \leftarrow \text{'Δεν ορίζεται'}$

Τέλος_αν

Εμφάνισε “Το ακέραιο μέρος της διαίρεσης του”, α , “με το”, $\beta-3$, “είναι”, Z

Δ. Για την επίλυση του προβλήματος της μέγιστης θερμοκρασίας μίας πόλης κατά την διάρκεια ενός μήνα (30 ημέρες) και της εύρεσης της μέσης θερμοκρασίας του μήνα, δόθηκε ο παρακάτω αλγόριθμος που διαβάζει τις θερμοκρασίες μίας πόλης και τις αποθηκεύει σε μονοδιάστατο πίνακα. Να γράψετε τον αριθμό της κάθε γραμμής και από δίπλα ποια τα **συντακτικά** και ποια τα **λογικά λάθη** του παρακάτω αλγορίθμου.

1. Αλγόριθμος 1ENA_ONOMA
2. $\max \leftarrow \Theta[1]$
3. $S \leftarrow 0$
4. Για i από 1 μέχρι 30
5. Διάβασε $\Theta[i]$
6. $S \leftarrow S + \Theta[i]$
7. $i \leftarrow i + 1$
8. Τέλος_επανάληψης
9. Για i από 2 μέχρι 30
10. Αν $\Theta[i] < \max$ τότε
11. $\Theta[i] \rightarrow \max$
12. Τέλος_αν
13. Τέλος_επανάληψης
14. $MO \leftarrow S / 30$
15. Εμφάνισε ‘Μέση Θερμοκρασία = ‘ MO ‘και μέγιστη = ‘ \max
16. Τέλος_Προγράμματος

ΘΕΜΑ 2°

Σε ένα καζίνο γίνεται διαγωνισμός “black jack”. Συμμετέχουν 10 παίκτες και κάθε παίκτης διαγωνίζεται σε 7 παρτίδες με διαφορετικό γκρουπιέρη σε κάθε παρτίδα. Να γίνει πρόγραμμα στην γλώσσα προγραμματισμού «ΓΛΩΣΣΑ» το οποίο:

1. Διαβάζει τα ονόματα των 10 παιχτών και των 7 γκρουπιέρηδων και τα αποθηκεύει σε κατάλληλους πίνακες.
2. Διαβάζει σε δισδιάστατο πίνακα ΑΠΟΤ[10,7] τα αποτελέσματα της κάθε παρτίδας κάθε παίκτη, ‘Κ’ για νίκη του παίκτη και ‘Χ’ για ήττα και κάνει έλεγχο εγκυρότητας δεδομένων.
3. Διαβάζει σε δισδιάστατο πίνακα ΠΟΣΑ[10,7] το ποσό που κέρδισε ή έχασε ο παίκτης σε κάθε παρτίδα (θεωρήστε διαφορετικά ποσά πονταρίσματος σε κάθε παρτίδα). Θεωρείστε ότι κάθε στοιχείο του πίνακα ΑΠΟΤ βρίσκεται σε αντιστοιχία με τα στοιχεία του πίνακα ΠΟΣΑ που βρίσκονται στην ίδια θέση.
4. Να εμφανίζει το όνομα κάθε παίκτη και την λέξη «ΚΕΡΔΟΣ» αν ο παίκτης συνολικά κέρδισε λεφτά ή την λέξη «ΖΗΜΙΑ» αν ο παίκτης συνολικά έχασε λεφτά.
5. Να δημιουργεί δύο νέους πίνακες με όνομα «ΚΕΡΔΙΣΜΕΝΟΙ» και «ΖΗΜΙΩΜΕΝΟΙ» με τα ονόματα των κερδισμένων και ζημιωμένων παιχτών αντίστοιχα
6. Εμφανίζει τα ονόματα των γκρουπιέρηδων οι οποίοι έχασαν από τους περισσότερους παίκτες ασχέτως του ποσού που πονταρίστηκε σε κάθε παρτίδα.

ΘΕΜΑ 3^ο

Συγχώνευση δύο ή περισσότερων ταξινομημένων πινάκων είναι η δημιουργία ενός τρίτου πίνακα που περιέχει τα στοιχεία των πρώτων ταξινομημένα. Να γίνει αλγόριθμος που δοθέντων δύο πινάκων με αύξουσα ταξινόμηση, A με M στοιχεία και B με N στοιχεία, δημιουργεί έναν τρίτο πίνακα Γ με M+N στοιχεία που περιέχει τα στοιχεία των 2 πρώτων ταξινομημένα κατά αύξουσα σειρά

ΛΥΣΕΙΣ ΘΕΜΑΤΩΝ

ΘΕΜΑ 1^ο

A : 1-Σ, 2-Λ, 3-Σ, 4-Λ, 5-Λ

```
B:      S ← 0
        i ← 1
        ΟΣΟ i <= 10 ΕΠΑΝΑΛΑΒΕ
            S ← S + 5
            i ← i + 1
        ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
        Εμφάνισε S, i
```

Γ: Το πρώτο αλγοριθμικό κριτήριο το οποίο δεν ικανοποιείται είναι αυτό της **καθοριστικότητας** καθώς δεν μπορεί κανείς να πει με ακρίβεια πως η μεταβλητή «Z» είναι αριθμητική ή αλφαριθμητική (στις 2 περιπτώσεις της δομής επιλογής). Το δεύτερο κριτήριο είναι η **αποτελεσματικότητα** καθώς το τελικό «Εμφάνισε» αναφέρει πως θα εμφανίσει ακέραιο μέρος της διαίρεσης 2 ακεραίων (πράξη DIV) ενώ η διαίρεση που γίνεται δεν υπολογίζει το ακέραιο μέρος άλλα ένα απλό πηλίκο. Συνεπώς δεν μας δίνει σωστό αποτέλεσμα. Τα υπόλοιπα τρία κριτήρια (είσοδος, έξοδος, περατότητα) ικανοποιούνται.

Δ :

Γραμμή	Τύπος λάθους	Αιτιολόγηση
1	Συντακτικό	Το όνομα ενός αλγορίθμου δεν μπορεί να ξεκινάει από αριθμό
2	Λογικό	Ο πίνακας Θ δεν έχει διαβαστεί ακόμα οπότε δεν ξέρουμε ποια θα είναι η πρώτη τιμή που θα πάρει
7	Λογικό	Το «i» μεταβάλλεται κατά ένα με την χρήση της εντολής «ΓΙΑ». Αν τοποθετηθεί η εντολή αυτή το i θα μεταβάλλεται κατά 2 και θα παραλείπονται θέσεις του πίνακα
10	Λογικό	Απαιτείται το σύμβολο ">" για την εύρεση μεγίστου
11	Συντακτικό	← είναι η σωστή μορφή εκχώρησης
12	Συντακτικό	Λείπει η κάτω παύλα: Τέλος_αν
15	Συντακτικό	Λείπουν τα κόμματα μεταξύ των μηνυμάτων και των μεταβλητών
16	Συντακτικό	Δεν έχουμε πρόγραμμα, απαιτείται η εντολή «Τέλος» και το όνομα του

Θέμα 2°

ΠΡΟΓΡΑΜΜΑ THEMA_2

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: i, j, K, L, P[7], max

ΠΡΑΓΜΑΤΙΚΕΣ: ΠΟΣΑ[10,7], S[10]

ΧΑΡΑΚΤΗΡΕΣ: ON[10], ΓΚΡ[7], ΑΠΟΤ[10,7], ΚΕΡΔΙΣΜΕΝΟΙ[10], ΖΗΜΙΩΜΕΝΟΙ[10]

ΑΡΧΗ

!-----Ερωτημα 1ο -----!

ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 10

ΓΡΑΨΕ 'ΔΩΣΤΕ ΤΟ ΟΝΟΜΑ ΤΟΥ ΠΑΙΧΤΗ', i

ΔΙΑΒΑΣΕ ON[i]

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 7

ΓΡΑΨΕ 'ΔΩΣΤΕ ΤΟ ΟΝΟΜΑ ΤΟΥ ΓΚΡΟΥΠΙΕΡΗ', i

ΔΙΑΒΑΣΕ ΓΚΡ[i]

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

!-----Ερωτημα 2ο και 3ο -----!

ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 10

ΓΙΑ j ΑΠΟ 1 ΜΕΧΡΙ 7

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ 'ΠΛΗΚΤΡΟΛΟΓΗΣΤΕ Κ ΓΙΑ ΝΙΚΗ, Χ ΓΙΑ ΗΤΤΑ ΤΟΥ ΠΑΙΧΤΗ ',ON[i], 'ΣΤΗΝ ΠΑΡΤΙΔΑ', j

ΔΙΑΒΑΣΕ ΑΠΟΤ[i,j]

ΜΕΧΡΙΣ_ΟΤΟΥ ΑΠΟΤ[i,j]='Κ' Η ΑΠΟΤ[i,j]='Χ'

ΓΡΑΨΕ 'ΠΛΗΚΤΡΟΛΟΓΗΣΤΕ ΠΟΣΟ ΠΟΝΤΑΡΙΣΜΑΤΟΣ ΤΟΥ ΠΑΙΧΤΗ',ON[i], 'ΣΤΗΝ ΠΑΡΤΙΔΑ', j

ΔΙΑΒΑΣΕ ΠΟΣΑ[i,j]

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

!-----Ερωτημα 4ο -----!

ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 10

S[i] <-- 0

ΓΙΑ j ΑΠΟ 1 ΜΕΧΡΙ 7

ΑΝ ΑΠΟΤ[i,j]= 'Κ' ΤΟΤΕ

S[i] <-- S[i] + ΠΟΣΑ[i,j]

ΑΛΛΙΩΣ

S[i] <-- S[i] - ΠΟΣΑ[i,j]

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 10

ΑΝ S[i] > 0 ΤΟΤΕ

ΓΡΑΨΕ ON[i], 'ΚΕΡΔΟΣ'

ΑΛΛΙΩΣ_ΑΝ S[i] < 0 ΤΟΤΕ

ΓΡΑΨΕ ON[i], 'ΖΗΜΙΑ'

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

!-----Ερωτημα 5ο -----!

K <-- 1

L <-- 1

ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 10

ΑΝ S[i] > 0 ΤΟΤΕ

ΚΕΡΔΙΣΜΕΝΟΙ[K] <-- ON[i]

```

    K <-- K + 1
ΑΛΛΙΩΣ_ΑΝ S[i] < 0 ΤΟΤΕ
    ΖΗΜΙΩΜΕΝΟΙ[L] <-- ΟΝ[i]
    L <-- L + 1
ΤΕΛΟΣ_ΑΝ
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

!-----Ερωτημα 6ο -----!
ΓΙΑ j ΑΠΟ 1 ΜΕΧΡΙ 7
    Ρ[j] <-- 0
    ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 10
        ΑΝ ΑΠΟΤ[i,j]= 'κ' ΤΟΤΕ
            Ρ[j] <-- Ρ[j] + 1
        ΤΕΛΟΣ_ΑΝ
    ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

max <-- Ρ[1]
ΓΙΑ j ΑΠΟ 2 ΜΕΧΡΙ 7
    ΑΝ Ρ[j] > max ΤΟΤΕ
        max <-- Ρ[j]
    ΤΕΛΟΣ_ΑΝ
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΙΑ j ΑΠΟ 1 ΜΕΧΡΙ 7
    ΑΝ Ρ[j] = max ΤΟΤΕ
        ΓΡΑΨΕ 'Όνομα Γκρουπιέρη με περισσότερες ήττες είναι :', ΓΚΡ[j]
    ΤΕΛΟΣ_ΑΝ
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

```

ΘΕΜΑ 3°

Τοποθετούμε στις πρώτες θέσεις του πίνακα Γ τα στοιχεία του πίνακα Α και στις επόμενες τα στοιχεία του πίνακα Β, και στην συνέχεια ταξινομούμε τον νέο πίνακα Γ

```

Αλγόριθμος ΣΥΓΧΩΝΕΥΣΗ
    Δεδομένα //Α, Β, Μ, Ν//
    Για i από 1 μέχρι Μ
        Γ[i] ← Α[i]
    Τέλος_επανάληψης

    Για i από Μ+1 μέχρι Μ+Ν
        Γ[i] ← Β[i-Μ]
    Τέλος_επανάληψης

    Για k από 2 μέχρι Μ+Ν
        Για i από Μ+Ν μέχρι k με_βήμα -1
            Αν Γ[i-1] < Γ[i] τότε
                temp ← Γ[i-1]
                Γ[i-1] ← Γ[i]
                Γ[i] ← temp
            Τέλος_αν
        Τέλος_επανάληψης
    Τέλος_επανάληψης
    Αποτελέσματα //Γ//
Τέλος ΣΥΓΧΩΝΕΥΣΗ

```

ΚΑΛΗ ΕΠΙΤΥΧΙΑ

Επιμέλεια Θεμάτων: Καρατόλιος Δημήτριος