

ΑΝΑΛΥΣΗ ΘΕΜΑΤΟΣ 2019-Δ (Υποπρόγραμμα)

Στην 27η **Βαλκανιάδα Πληροφορικής** που θα διεξαχθεί στην Αθήνα τον Σεπτέμβριο του 2019 , **συμμετέχουν 40 μαθητές**. Κάθε **μαθητής** παίρνει έναν **κωδικό από 1 έως και 40** , ο οποίος αντιστοιχεί στη σειρά που δήλωσε συμμετοχή. Κάθε **μαθητής** καλείται να επιλύσει **έξι προβλήματα** . Για κάθε πρόβλημα αναπτύσσει τη **λύση** του σε μία γλώσσα προγραμματισμού και την υποβάλλει για **βαθμολόγηση**. Η λύση **βαθμολογείται σε ακέραια κλίμακα από 0 έως 100**.

Κατά τη **διάρκεια** του διαγωνισμού κάθε **μαθητής** και για κάθε πρόβλημα μπορεί να **υποβάλλει τη λύση** του **όσες φορές θέλει**.

Να αναπτύξετε πρόγραμμα σε ΓΛΩΣΣΑ το οποίο:

Δ1. Να περιλαμβάνει κατάλληλο τμήμα δηλώσεων.

ΠΡΟΓΡΑΜΜΑ ΘΕΜΑ_Δ

ΣΤΑΘΕΡΕΣ

N = 40

M = 6

ΜΕΤΑΒΛΗΤΕΣ

ΑΡΧΗ

Δ2. Να διαβάζει επαναληπτικά τα **ονόματα** των μαθητών και να τα καταχωρίζει στον Πίνακα **ΟΝ[40]**.

Επίσης, να **αρχικοποιεί με την τιμή 0** όλα τα στοιχεία του Πίνακα **ΒΑΘ[40,6]**, ο οποίος θα περιέχει τη **βαθμολογία** κάθε μαθητή για κάθε πρόβλημα.

ΓΙΑ **I** **ΑΠΟ** 1 **ΜΕΧΡΙ** **N**

ΔΙΑΒΑΣΕ **ΟΝ** [**I**]

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΙΑ **I** **ΑΠΟ** 1 **ΜΕΧΡΙ** **N**

ΓΙΑ **j** **ΑΠΟ** 1 **ΜΕΧΡΙ** **M**

ΒΑΘ[**I**, **j**] <- 0

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

Δ3. Κάθε φορά που μία **λύση** προβλήματος **υποβάλλεται και βαθμολογείται**, το πρόγραμμα να **διαβάζει τον κωδικό** του μαθητή (από 1 έως και 40), τον **αριθμό του προβλήματος** (από 1 έως και 6) και τη **βαθμολογία** του (από 0 έως και 100).

Η **βαθμολογία** να **καταχωρίζεται** στην αντίστοιχη θέση του Πίνακα **ΒΑΘ[40,6]** μόνο αν είναι **μεγαλύτερη** από τη **βαθμολογία** που είναι ήδη καταχωρισμένη .

Για τον **τερματισμό** της εισαγωγής δεδομένων το πρόγραμμα να εμφανίζει το μήνυμα «**Υπάρχει νέα λύση προβλήματος; ΝΑΙ / ΟΧΙ**» . Αν εισαχθεί η τιμή «**ΟΧΙ**» , να **τερματίζεται η εισαγωγή δεδομένων**.

ΠΛΗΘΟΣ ΜΑΘΗΤΩΝ :40

ΠΛΗΘΟΣ ΠΡΟΒΛΗΜΑΤΩΝ :6

ΚΜ : ΚΩΔΙΚΟΣ ΜΑΘΗΤΗ (1- 40)

ΑΠ : ΑΡΙΘΜΟΣ ΠΡΟΒΛΗΜΑΤΟΣ (1 - 6)

ΒΑΘΜΟΛΟΓΙΑ : **B** (1 - 100)

ΟΝ[40] ΠΙΝΑΚΑΣ ΟΝΟΜΑΤΑ ΜΑΘΗΤΩΝ

ΒΑΘ[40,6] ΠΙΝΑΚΑΣ ΒΑΘΜΟΛΟΓΙΕΣ ΜΑΘΗΤΩΝ

Η βαθμολογία να καταχωρίζεται στην αντίστοιχη θέση του Πίνακα ΒΑΘ[40,6] μόνο αν είναι **μεγαλύτερη από τη βαθμολογία** που είναι ήδη καταχωρημένη

ΑΝ (B > ΒΑΘ[ΚΜ, ΑΠ]) **ΤΟΤΕ**

ΒΑΘ[ΚΜ, ΑΠ] <- B

ΤΕΛΟΣ_ΑΝ

Για τον τερματισμό της εισαγωγής δεδομένων το πρόγραμμα να εμφανίζει το μήνυμα «Υπάρχει νέα λύση προβλήματος; ΝΑΙ / ΟΧΙ» . Αν εισαχθεί η τιμή «ΟΧΙ» να τερματίζεται η εισαγωγή δεδομένων

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ

ΔΙΑΒΑΣΕ ΚΜ, ΑΠ, B

ΑΝ (B > ΒΑΘ[ΚΜ, ΑΠ]) **ΤΟΤΕ**

ΒΑΘ[ΚΜ, ΑΠ] <- B

ΤΕΛΟΣ_ΑΝ

ΓΡΑΨΕ 'Υπάρχει νέα λύση προβλήματος; ΝΑΙ / ΟΧΙ'

ΔΙΑΒΑΣΕ ΑΠΑΝΤΗΣΗ

ΜΕΧΡΙΣ_ΟΤΟΥ (ΑΠΑΝΤΗΣΗ = 'ΟΧΙ')

Δ4 Υπολογίζει και να καταχωρίζει στον Πίνακα **ΣΒ[40]** τα **αθροίσματα των βαθμολογιών κάθε μαθητή στα έξι προβλήματα**

Για τον σκοπό αυτό να καλεί μόνο μια φορά το υποπρόγραμμα με όνομα **ΥΣΒ**

ΚΑΛΕΣΕ ΥΣΒ(ΒΑΘ, ΣΒ)

Το υποπρόγραμμα ΥΣΒ δέχεται ως **είσοδο τον Πίνακα ΒΑΘ[40,6]** και να επιστρέφει ως **έξοδο συμπληρωμένο τον Πίνακα ΣΒ[40]**.

ΠΑΡΑΔΕΙΓΜΑ

ΟΝΟΜΑΤΑ	ΒΑΘΜΟΛΟΓΙΕΣ		ΣΥΝΟΛΟ (ΣΒ)
A	12	13	25
N	10	11	21
Φ	10	15	25
B	17	11	28

ΔΙΑΔΙΚΑΣΙΑ ΥΣΒ(ΒΑΘ, ΣΒ)

ΣΤΑΘΕΡΕΣ

N = 40

M = 6

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: I, J, ΒΑΘ[N, M], ΣΒ[N]

ΑΡΧΗ

ΓΙΑ I **ΑΠΟ** 1 **ΜΕΧΡΙ** N

ΣΒ[I] <- 0

ΓΙΑ J **ΑΠΟ** 1 **ΜΕΧΡΙ** M

ΣΒ[I] <- ΣΒ[I] + ΒΑΘ[I, J]

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΤΕΛΟΣ ΔΙΑΔΙΚΑΣΙΑΣ

Δ5. Να εμφανίζει τα **ονόματα των μαθητών ταξινομημένων** σύμφωνα με τη **συνολική τους βαθμολογία** σε **φθίνουσα βαθμολογική σειρά**.

Σε περίπτωση μαθητών με την **ίδια βαθμολογία**, τα **ονόματά τους** να εμφανίζονται με **αλφαβητική σειρά**.

Σημειώσεις

α) Δεν απαιτούνται έλεγχοι εγκυρότητας τιμών .

β) Να θεωρήσετε ότι θα δοθεί τουλάχιστον μια λύση προβλήματος από έναν μαθητή.

ΠΑΡΑΔΕΙΓΜΑ

ΟΝΟΜΑΤΑ	ΒΑΘΜΟΛΟΓΙΕΣ		ΣΥΝΟΛΟ (ΣΒ)
A	12	13	25
N	10	11	21
Φ	10	15	25
B	17	11	28

ΤΑΞΙΝΟΜΗΣΗ

ΟΝΟΜΑΤΑ	ΣΥΝΟΛΟ (ΣΒ)
B	28
A	25
Φ	25
N	21

ΓΙΑ Ι ΑΠΟ 2 ΜΕΧΡΙ Ν

ΓΙΑ j ΑΠΟ Ν ΜΕΧΡΙ Ι ΜΕ_ΒΗΜΑ -1

ΑΝ (ΣΒ[j] > ΣΒ[j - 1]) ΤΟΤΕ

Κ <- ΣΒ[j]

ΣΒ[j] <- ΣΒ[j - 1]

ΣΒ[j - 1] <- Κ

Λ <- ΟΝ[j]

ΟΝ[j] <- ΟΝ[j - 1]

ΟΝ[j - 1] <- Λ

ΑΛΛΙΩΣ_ΑΝ (ΣΒ[j] = ΣΒ[j - 1]) ΤΟΤΕ

ΑΝ (ΟΝ[j] < ΟΝ[j - 1]) ΤΟΤΕ

Λ <- ΟΝ[j]

ΟΝ[j] <- ΟΝ[j - 1]

ΟΝ[j - 1] <- Λ

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΙΑ Ι ΑΠΟ 1 ΜΕΧΡΙ Ν

ΓΡΑΨΕ ΟΝ [Ι] '

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ