

## Θέμα 1ο.

1. Χαρακτηρίστε κάθε μιας από τις ακόλουθες προτάσεις, σαν σωστή (Σ) ή λάθος(Λ) (μονάδες 10)
- Η παραγοντοποίηση ενός θετικού ακεραίου αριθμού Ν, σε γινόμενο πρώτων παραγόντων, αποτελεί πρόβλημα βελτιστοποίησης.
  - Κάθε μεταβλητή παίρνει τιμή μόνο με την εντολή εκχώρησης.
  - Η χρήση πινάκων είναι ένας βολικός τρόπος για τη διαχείριση πολλών δεδομένων διαφορετικού τύπου.
  - Ο τρόπος σύνταξης μιας διαδικασίας διαφέρει απ' τον τρόπο σύνταξης μιας συνάρτησης, ενώ ο τρόπος κλήσης τους είναι ο ίδιος.
  - Όταν σ'ένα υποπρόγραμμα έχουμε παράμετρο πίνακα, τότε η τυπική και η αντίστοιχη πραγματική παράμετρος, πρέπει να είναι ίδιου τύπου και ίδιου μεγέθους.

## 2.

- i. Να γράψετε σε Γλώσσα την εντολή εκχώρησης που αντιστοιχεί στην παρακάτω αριθμητική

παράσταση.  $x_1 = \frac{-\beta + \sqrt{\Delta}}{2a}$

(μονάδες 3)

- ii. Να γράψετε σε ΓΛΩΣΣΑ τη λογική έκφραση (συνθήκη), η οποία ελέγχει τις τιμές δύο μεταβλητών χ,ψ όπως περιγράφεται στην πρόταση:

το χ είναι μικρότερο του 10 ή το ψ μικρότερο του 10, αλλά όχι και τα δύο συγχρόνως.

(Μονάδες 3)

3. Να αντιστοιχίσετε κάθε τμήμα ψευδοκώδικα της Στήλης Α με το ισοδύναμό του τμήμα προγράμματος της Στήλης Β.

(Μονάδες 9)

	ΣΤΗΛΗ Α		ΣΤΗΛΗ Β
1.	10 ΔΙΑΒΑΣΕ Α 20 ΑΝ Α<=0 ΤΟΤΕ ΓΟΤΟ 50 30 ΓΡΑΨΕ 'ΘΕΤΙΚΟ' 40 ΓΟΤΟ 10 50 ΓΡΑΨΕ 'ΜΗ ΘΕΤΙΚΟ' 60 ΤΕΛΟΣ	α.	ΔΙΑΒΑΣΕ Α ΌΣΟ Α<=0 ΕΠΑΝΑΛΑΒΕ ΓΡΑΨΕ 'ΜΗ ΘΕΤΙΚΟ' ΔΙΑΒΑΣΕ Α ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ ΓΡΑΨΕ 'ΘΕΤΙΚΟ' ΤΕΛΟΣ
2.	10 ΔΙΑΒΑΣΕ Α 20 ΑΝ Α<=0 ΤΟΤΕ ΓΟΤΟ 50 30 ΓΡΑΨΕ 'ΘΕΤΙΚΟ' 40 ΓΟΤΟ 60 50 ΓΡΑΨΕ 'Μ Η ΘΕΤΙΚΟ' 60 ΤΕΛΟΣ	β.	ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ ΔΙΑΒΑΣΕ Α ΑΝ Α>0 ΤΟΤΕ ΓΡΑΨΕ 'ΘΕΤΙΚΟ' ΤΕΛΟΣ_ΑΝ ΜΕΧΡΙΣ_ΌΤΟΥ Α<=0 ΓΡΑΨΕ 'ΜΗ ΘΕΤΙΚΟ' ΤΕΛΟΣ
3.	10 ΔΙΑΒΑΣΕ Α 20 ΑΝ Α>0 ΤΟΤΕ ΓΟΤΟ 50 30 ΓΡΑΨΕ 'ΜΗ ΘΕΤΙΚΟ' 40 ΓΟΤΟ 10 50 ΓΡΑΨΕ 'ΘΕΤΙΚΟ' 60 ΤΕΛΟΣ	γ.	ΔΙΑΒΑΣΕ Α ΑΝ Α>0 ΤΟΤΕ ΓΡΑΨΕ 'ΘΕΤΙΚΟ' ΑΛΛΙΩΣ ΓΡΑΨΕ 'ΜΗ ΘΕΤΙΚΟ' ΤΕΛΟΣ_ΑΝ ΤΕΛΟΣ

4. Να γράψετε αλγόριθμο που διαβάζει δύο αριθμούς και τυπώνει τον μεγαλύτερο, με τη χρήση μόνο **μίας απλής** εντολής επιλογής. Θεωρήστε δεδομένο ότι δίνονται δύο άνισοι αριθμοί. (Μονάδες 9)
5. Δίνεται η παρακάτω συνάρτηση η οποία υλοποιεί τον αλγόριθμο της σειριακής αναζήτησης ακέрайου αριθμού σε πίνακα ακεραίων, στον οποίο θεωρούμε ότι μπορεί να υπάρχει το πολύ μία φορά. Η αναζήτηση θα σταματήσει μόλις βρεθεί ο αριθμός ή όταν ελεγχθούν όλες οι θέσεις του πίνακα. Όταν βρεθεί ο αριθμός η συνάρτηση θα επιστρέφει την θέση του μέσα στον πίνακα, διαφορετικά την τιμή 0. Η συνάρτηση έχει τυπικές παραμέτρους πίνακα 1000 ακεραίων θέσεων και μια ακέрайα μεταβλητή, την τιμή της οποίας θα αναζητήσει.

ΣΥΝΑΡΤΗΣΗ Sequential\_Search(table,key):ΑΚΕΡΑΙΑ

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ:table[1000],key,i,position

ΛΟΓΙΚΕΣ:done

ΑΡΧΗ

Done←ΨΕΥΔΗΣ

Position←0

I←1

ΌΣΟ (done=ΨΕΥΔΗΣ)ΚΑΙ(I <=1000)ΕΠΑΝΑΛΑΒΕ

ΑΝ table[I]=key ΤΟΤΕ

Done←ΑΛΗΘΗΣ

Position← I

ΑΛΛΙΩΣ

I←I +1

ΤΕΛΟΣ\_ΑΝ

ΤΕΛΟΣ\_ΕΠΑΝΑΛΗΨΗΣ

Sequential\_Search ← position

ΤΕΛΟΣ\_ΣΥΝΑΡΤΗΣΗΣ

Η παρακάτω Συνάρτηση αν συμπληρωθεί σωστά υλοποιεί την ίδια διεργασία με την προηγούμενη, στηριζόμενη όμως σε πιο αποδοτικό αλγόριθμο (ταχύτερη εκτέλεση).

ΣΥΝΑΡΤΗΣΗ Sequential\_Search(table,key):ΑΚΕΡΑΙΑ

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ:table[1000],key,i

ΑΡΧΗ

I←\_\_\_\_\_

ΑΡΧΗ\_ΕΠΑΝΑΛΗΨΗΣ

I←I +1

ΜΕΧΡΙΣ\_ΌΤΟΥ (\_\_\_\_\_) Ή (I =1000)

ΑΝ \_\_\_\_\_ ΤΟΤΕ

Sequential\_Search ← I

ΑΛΛΙΩΣ

Sequential\_Search ← 0

ΤΕΛΟΣ\_ΑΝ

ΤΕΛΟΣ\_ΣΥΝΑΡΤΗΣΗΣ

Να συμπληρώσετε τα κενά \_\_\_\_\_.

(Μονάδες 6)

## Θέμα 2ο.

1. Δίνεται το παρακάτω πρόγραμμα το οποίο καλεί τα 3 υποπρογράμματα που ακολουθούν:

ΠΡΟΓΡΑΜΜΑ ΠΡΑΞΕΙΣ

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: Α,Β,Γ,Χ,Υ,Ρ

ΑΡΧΗ

ΑΡΧΗ\_ΕΠΤΑΝΑΛΗΨΗΣ

ΔΙΑΒΑΣΕ Α,Β

ΜΕΧΡΙΣ\_ΟΤΟΥ (Α>0)ΚΑΙ(Β>0)

Γ←Π(Α,Β)

ΚΑΛΕΣΕ Δ(Α,Β,Χ,Υ)

Ρ←ΔΝ(Α,Β)

ΓΡΑΨΕ Γ,Χ,Υ,Ρ

ΤΕΛΟΣ\_ΠΡΟΓΡΑΜΜΑΤΟΣ

ΣΥΝΑΡΤΗΣΗ Π(Α1,Α2):ΑΚΕΡΑΙΑ

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: Α1,Α2,Ι,Σ

ΑΡΧΗ

Σ←0

ΓΙΑ Ι ΑΠΟ 1 ΜΕΧΡΙ Α1

Σ←Σ+Α2

ΤΕΛΟΣ\_ΕΠΤΑΝΑΛΗΨΗΣ

Π←Σ

ΤΕΛΟΣ\_ΣΥΝΑΡΤΗΣΗΣ

ΔΙΑΔΙΚΑΣΙΑ Δ(Α1,Α2,Π,Υ)

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: Α1,Α2,Π,Υ

ΑΡΧΗ

Π←0

Υ←Α1

ΟΣΟ Υ>=Α2 ΕΠΤΑΝΑΛΑΒΕ

Π←Π+1

Υ←Υ-Α2

ΤΕΛΟΣ\_ΕΠΤΑΝΑΛΗΨΗΣ

ΤΕΛΟΣ\_ΔΙΑΔΙΚΑΣΙΑΣ

ΣΥΝΑΡΤΗΣΗ ΔΝ(Α1,Α2):ΑΚΕΡΑΙΑ

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: Α1,Α2,Ι,Γ

ΑΡΧΗ

Γ←1

ΓΙΑ Ι ΑΠΟ 1 ΜΕΧΡΙ Α2

Γ←Γ\*Α1

ΤΕΛΟΣ\_ΕΠΤΑΝΑΛΗΨΗΣ

ΔΝ←Γ

ΤΕΛΟΣ\_ΣΥΝΑΡΤΗΣΗΣ

Να συμπληρώσετε, τις τέσσερις(4) εντολές εκχώρησης του παρακάτω προγράμματος που περιέχουν κενά, βάζοντας στη θέση των κενών με τους κατάλληλους αριθμητικούς τελεστές, έτσι ώστε το πρόγραμμα να δίνει ισοδύναμο αποτέλεσμα με το αρχικό.

ΠΡΟΓΡΑΜΜΑ ΠΡΑΞΕΙΣ

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: Α,Β,Γ,Χ,Υ,Ρ

ΑΡΧΗ

ΑΡΧΗ\_ΕΠΤΑΝΑΛΗΨΗΣ

ΔΙΑΒΑΣΕ Α,Β

ΜΕΧΡΙΣ\_ΟΤΟΥ (Α>0)ΚΑΙ(Β>0)

Γ←Α \_\_\_\_\_ Β

Χ←Α \_\_\_\_\_ Β

Υ←Α \_\_\_\_\_ Β

Ρ←Α \_\_\_\_\_ Β

ΓΡΑΨΕ Γ,Χ,Υ,Ρ

ΤΕΛΟΣ\_ΠΡΟΓΡΑΜΜΑΤΟΣ

(Μονάδες 12)

2. Δίνεται το παρακάτω πρόγραμμα το οποίο ελέγχει αν η συνάρτηση ΑΛΑ\_ΡΩΣΙΚΑ, η οποία υλοποιεί τον αλγόριθμο του Ρώσικου πολλαπλασιασμού, υπολογίζει σωστά το γινόμενο 2 θετικών ακεραίων αριθμών.
- |                           |                                     |
|---------------------------|-------------------------------------|
| ΠΡΟΓΡΑΜΜΑ ΕΛΕΓΧΟΣ         | ΣΥΝΑΡΤΗΣΗ ΑΛΑ_ΡΩΣΙΚΑ(A1,A2):ΑΚΕΡΑΙΑ |
| ΜΕΤΑΒΛΗΤΕΣ                | ΜΕΤΑΒΛΗΤΕΣ                          |
| ΑΚΕΡΑΙΕΣ: Α,Β,Γ           | ΑΚΕΡΑΙΕΣ: Α1,Α2,Σ                   |
| ΑΡΧΗ                      | ΑΡΧΗ                                |
| ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ           | Σ ← 0                               |
| ΔΙΑΒΑΣΕ Α,Β               | ΟΣΟ Α2 > 0 ΕΠΑΝΑΛΑΒΕ                |
| ΜΕΧΡΙΣ_ΟΤΟΥ (Α>0)ΚΑΙ(Β>0) | ΑΝ Α2 MOD 2 = 1 ΤΟΤΕ                |
| Γ ← ΑΛΑ_ΡΩΣΙΚΑ(Α,Β)       | Σ ← Σ + Α1                          |
| ΑΝ Α * Β = Γ ΤΟΤΕ         | ΤΕΛΟΣ_ΑΝ                            |
| ΓΡΑΨΕ 'ΣΩΣΤΟ ΓΙΝΟΜΕΝΟ'    | Α1 ← Α1 * 2                         |
| ΑΛΛΙΩΣ                    | Α2 ← Α2 DIV 2                       |
| ΓΡΑΨΕ 'ΛΑΘΟΣ ΓΙΝΟΜΕΝΟ'    | ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ                    |
| ΤΕΛΟΣ_ΑΝ                  | ΑΛΑ_ΡΩΣΙΚΑ ← Σ                      |
| ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ        | ΤΕΛΟΣ_ΣΥΝΑΡΤΗΣΗΣ                    |
- i. Να γράψετε διαδικασία ισοδύναμη με τη συνάρτηση. (Μονάδες 4)
- ii. Να ξαναγράψετε το πρόγραμμα ώστε να χρησιμοποιεί την διαδικασία. (Μονάδες 4)

**Θέμα 3ο.** Το γνωστό παιχνίδι «ΚΡΕΜΑΛΑ» θα μπορούσε να υλοποιηθεί σε υπολογιστή ως εξής: Η άγνωστη λέξη θα εμφανίζεται στην οθόνη του υπολογιστή με τη γνωστή μορφή, δηλαδή θα εμφανίζονται το πρώτο και το τελευταίο γράμμα και ενδιάμεσα παύλες (π.χ. Α - - - - - Σ). Ο χρήστης-παίκτης θα προσπαθεί να τη βρει δίνοντας ένα-ένα τα γράμματα που πιστεύει ότι περιέχει. Σε κάθε προσπάθεια το πρόγραμμα θα επανεμφανίζει τη λέξη με τις παύλες, ενημερωμένη με τα γράμματα που πιθανόν βρέθηκαν. Το παιχνίδι θα τελειώνει με νικητή το χρήστη όταν βρίσκει τη λέξη, δίνοντάς του τη δυνατότητα να δώσει μέχρι και 7 λανθασμένα γράμματα. Να αναπτύξετε πρόγραμμα σε «ΓΛΩΣΣΑΣ» το οποίο :

1. Περιλαμβάνει τμήμα δηλώσεων μεταβλητών. (Μονάδες 2)
2. Καλεί προκατασκευασμένο υποπρόγραμμα απ' τη βιβλιοθήκη της «ΓΛΩΣΣΑΣ», το οποίο ονομάζεται ΔΗΜΙΟΥΡΓΙΑ\_ΛΕΞΗΣ(P) και έχει τυπική παράμετρο πίνακα P[8] χαρακτήρων. Ο πίνακας που επιστρέφει το υποπρόγραμμα περιέχει σε κάθε μια απ' τις οκτώ θέσεις του ένα κεφαλαίο ελληνικό γράμμα, δηλαδή συνολικά μια λέξη οκτώ γραμμάτων. (Το υποπρόγραμμα αυτό, δεν θα το κατασκευάσετε, απλώς θα το καλέσετε μια φορά στην αρχή για να σας δημιουργήσει την άγνωστη λέξη, την οποία επιλέγει τυχαία από ένα αρχείο με πολλές λέξεις οκτώ γραμμάτων.) (Μονάδες 1)
3. Πριν απ' το διάβασμα κάθε γράμματος που θα δίνει ο χρήστης, το πρόγραμμα θα καλεί προκατασκευασμένο υποπρόγραμμα απ' τη βιβλιοθήκη της «ΓΛΩΣΣΑΣ», το οποίο ονομάζεται ΓΡΑΨΕ\_ΛΕΞΗ(P) με τυπική παράμετρο πίνακα P[8] χαρακτήρων το οποίο θα εμφανίζει στην οθόνη τους χαρακτήρες που περιέχει ο πίνακας. (Το υποπρόγραμμα αυτό δεν θα το κατασκευάσετε, απλώς θα το καλείτε κάθε φορά για να εμφανίζει την άγνωστη λέξη.) (Μονάδες 12)
4. Το παιχνίδι θα τελειώνει όταν βρεθεί η λέξη, εμφανίζοντας τις προσπάθειες που χρειάστηκε ο χρήστης για να τη βρει, είτε αν δοθούν 8 λανθασμένα γράμματα, οπότε θα αποκαλύπτει την άγνωστη λέξη. (μ 5)  
Διευκρινίσεις:  
Τα γράμματα που πρέπει να βρεθούν είναι 6.

Δεν χρειάζεται έλεγχος των δεδομένων που εισάγονται)

Θεωρείστε ότι ο χρήστης δεν ξαναδίνει το ίδιο γράμμα δεύτερη φορά.

Η λέξη μπορεί να έχει πάνω από μια φορά το ίδιο γράμμα.

Αν δοθεί γράμμα που υπάρχει περισσότερες από μία φορές στη λέξη, τότε θεωρείστε ότι έχουν βρεθεί με μια προσπάθεια τόσα γράμματα όσα περιέχει η λέξη.

- Θέμα 4ο.** Το Αριθμολαχείο (Λόττο) είναι ένα παιχνίδι τύχης με επιλογή αριθμών. Αφορά στην ακριβή πρόβλεψη έξι (6) διαφορετικών αριθμών που κληρώνονται από μία σειρά σαράντα εννέα (49) αριθμών, από το 1 μέχρι και το 49. Μέχρι σήμερα έχουν γίνει 2300 κληρώσεις. Να αναπτύξετε αλγόριθμο ο οποίος δίνει τις εξής στατιστικές πληροφορίες:
1. αρχικά διαβάσει τις εξάδες αριθμών που έχουν κληρωθεί μέχρι σήμερα και τις αποθηκεύει σε πίνακα  $\Lambda[2300,6]$  (Δεν απαιτείται έλεγχος εγκυρότητας των δεδομένων.)
  2. Εμφανίζει τους 6 αριθμούς κάθε κλήρωσης, στην οποία κληρώθηκαν έξι ζυγοί ή έξι μονοί αριθμοί, διαφορετικά αν δεν συνέβη ποτέ αυτό, να εμφανίζει σχετικό μήνυμα.
  3. Εμφανίζει τον αριθμό ή τους αριθμούς, αν είναι περισσότεροι του ενός, που έχουν καθυστερήσει περισσότερο να κληρωθούν.
  4. Βρίσκει τους 6 αριθμούς που έχουν κληρωθεί τις λιγότερες φορές μέχρι σήμερα. Να τους εμφανίζει με αύξουσα διάταξη ως προς το πλήθος κληρώσεών τους.
  5. Να βρίσκει πόσοι το πολύ απ' τους 6 αριθμούς που έχουν κληρωθεί τις περισσότερες φορές μέχρι σήμερα, έχουν κληρωθεί όλοι μαζί σε μια κλήρωση.
- Σημείωση: Θεωρείστε ότι και οι 49 αριθμοί έχουν διαφορετικό πλήθος κληρώσεων μεταξύ τους.