# Ε.Α.Π./ΕΙΣΑΓΩΓΗ ΣΤΗΝ ΠΛΗΡΟΦΟΡΙΚΗ

**2η ΓΡΑΠΤΗ ΕΡΓΑΣΙΑ**

**ΑΚΑΔΗΜΑΪΚΟΥ ΕΤΟΥΣ 2022-2023**

**2ος Τόμος**

**ΤΕΧΝΙΚΕΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ**

**19/11/2022**

**Ημερομηνία παράδοσης εργασίας: Κυριακή 08/01/2023**

**Καταληκτική ημερομηνία παραλαβής: Τετάρτη 11/01/2023 [[1]](#footnote-1)**

**Ημερομηνία ανάρτησης ενδεικτικών λύσεων: Σάββατο 14/01/2023**

**Καταληκτική ημερομηνία αποστολής σχολίων στους φοιτητές: Πέμπτη 26/01/2023**

**ΥΠΟΕΡΓΑΣΙΑ 1.** (βαθμοί 30)

Αλγόριθμοι

**ΥΠΟΕΡΓΑΣΙΑ 2.** (βαθμοί 35)

Πίνακες, Ταξινόμηση

**ΥΠΟΕΡΓΑΣΙΑ 3.** (βαθμοί 35)

Πίνακες, Υποπρογράμματα, Διαδικασίες

**ΣΥΝΟΛΟ** **(βαθμοί 100)**

**ΥΠΟΕΡΓΑΣΙΑ 1. (βαθμοί 30)**

**Α.** Να γράψετε (α) αλγόριθμο με τη μορφή ψευδοκώδικα και (β) πρόγραμμα σε γλώσσα προγραμματισμού C για τον υπολογισμό των εξής αθροισμάτων:

**1)** *S1 = 1 – 2 + 3 – 4 ... ± N*

**2)** *S2 = 1 + 2! + 3! + 4! ... + N!,* όπου *Ν! = 1·2·3·...·Ν*

Το Ν θα πρέπει να εισάγεται από το πληκτρολόγιο και να είναι θετικός ακέραιος αριθμός μικρότερος του 10. Να εφαρμόσετε αμυντικό προγραμματισμό για την εισαγωγή του Ν.

**Β.** Ένας φοιτητής του ΕΑΠ έχει βαθμολογηθεί σε 10 εργασίες που έχει παραδώσει και θέλει να υπολογίσει τον μέσο όρο τους, τη διαφορά της μέγιστης βαθμολογίας που έχει λάβει από τον μέσο όρο του, καθώς και τη διαφορά της ελάχιστης βαθμολογίας που έχει λάβει από τον μέσο όρο του. Για να τον βοηθήσετε:

(α) να γράψετε έναν αλγόριθμο με τη μορφή ψευδοκώδικα που θα κάνει τα ακόλουθα:

1. Θα διαβάζει τους βαθμούς των δέκα εργασιών. Θεωρήστε ότι αυτοί οι βαθμοί είναι πραγματικοί αριθμοί από 0 μέχρι και 10 (δεν χρειάζεται να εφαρμόσετε αμυντικό προγραμματισμό).
2. Θα υπολογίζει και θα τυπώνει τον μέσο όρο των βαθμών.
3. Θα υπολογίζει τον ελάχιστο βαθμό του και θα τυπώνει τη διαφορά του από τον μέσο όρο.
4. Θα υπολογίζει τον μέγιστο βαθμό του και θα τυπώνει τη διαφορά του από τον μέσο όρο.

(β) να γράψετε πρόγραμμα σε γλώσσα προγραμματισμού C που θα υπολογίζει τα παραπάνω.

**ΥΠΟΕΡΓΑΣΙΑ 2. (βαθμοί 35)**

Ένας μονοδιάστατος πίνακας *Ν* *διαφορετικών* μεταξύ τους *μη αρνητικών* ακεραίων ονομάζεται *περιορισμένου εύρους R* αν το εύρος των στοιχείων του είναι μικρότερο από *R* (εάν, δηλαδή, η διαφορά μεταξύ του μεγίστου και του ελαχίστου στοιχείου είναι μικρότερη από την τιμή του *R*). Για παράδειγμα, ένας πίνακας που περιέχει τα οκτώ στοιχεία 900, 800, 750, 721, 880, 850, 840, 895 είναι *περιορισμένου εύρους R* για *R* = 200 καθώς η διαφορά του μεγίστου, 900, και του ελαχίστου, 721 είναι ίση με 179, η οποία είναι μικρότερη από την τιμή *R* = 200.

Ένας πίνακας δεδομένων που είναι *περιορισμένου εύρους R* μπορεί να ταξινομηθεί από οποιαδήποτε από τις γνωστές μεθόδους ταξινόμησης. Ωστόσο, ειδικά για τους πίνακες αυτούς μπορεί να χρησιμοποιηθεί η εξής, ταχύτερη, μέθοδος μέσω των παρακάτω βημάτων:

1) Κατασκευάζουμε έναν βοηθητικό πίνακα, R + 1 θέσεων, στον οποίο θα τεθεί η τιμή 1 στις θέσεις εκείνες που προκύπτουν από τους αριθμούς του αρχικού πίνακα δεδομένων μειωμένους κατά τον ελάχιστο από αυτούς, ενώ στις υπόλοιπες θέσεις θα τεθεί η τιμή 0.

Για τα στοιχεία του παραδείγματός μας έχουμε R = 200. Κατά συνέπεια, ο βοηθητικός πίνακας θα έχει 201 θέσεις. Θα περιέχει παντού την τιμή 0 εκτός από τις θέσεις που αντιστοιχούν σε κάποιο από τα δεδομένα μας αφού έχει αφαιρεθεί από τον καθένα ο min = 721. Στο παράδειγμά μας τα δεδομένα μείον τον ελάχιστο αριθμό (ίσο με 721) θα μας δώσουν τα εξής στοιχεία:

900 - 721, 800 - 721, 750 - 721, 721 - 721, 880 - 721, 850 - 721, 840 - 721, 895 – 721

ή

179, 79, 29, 0, 159, 129, 119, 174

ή (σε αύξουσα σειρά)

0, 29, 79, 119, 129, 159, 174, 179

Κατά συνέπεια, ο βοηθητικός πίνακας θα περιέχει παντού την τιμή 0 εκτός από τις θέσεις 179, 79, 29, 0, 159, 129, 119, 174, όπου θα περιέχει την τιμή 1 (αν, όμως, οι θέσεις του πίνακα αριθμούνται ξεκινώντας από το 1, οι θέσεις τροποποιούνται ανάλογα, δηλαδή αυξάνονται απλά κατά 1). Σημειώνεται ότι αυτές οι θέσεις είναι διαφορετικές μεταξύ τους, αφού διαφορετικοί είναι και οι αρχικοί αριθμοί άρα και η διαφορά τους από τον ελάχιστο.

2) Στη συνέχεια, σαρώνουμε τον βοηθητικό πίνακα από την αρχή προς το τέλος και για κάθε θέση *i* που περιέχει την τιμή 1, αποθηκεύουμε στην επόμενη θέση του αρχικού πίνακα, ξεκινώντας από την αρχή του, τον αριθμό i + min, ο οποίος μας δίνει ένα από τα αρχικά μας δεδομένα στη σωστή θέση του (αν, όμως, οι θέσεις του πίνακα αριθμούνται ξεκινώντας από το 1, τροποποιούμε κατάλληλα τον αριθμό αυτό αφαιρώντας το 1).

Συνεπώς, στο παράδειγμά μας, ο βοηθητικός πίνακας θα περιέχει το 1 στις θέσεις

0, 29, 79, 119, 129, 159, 174, 179

ενώ στις υπόλοιπες θέσεις θα περιέχει το 0.

Διατρέχουμε τον πίνακα αυτό προσθέτοντας στον *αύξοντα αριθμό* κάθε *θέσης* του που περιέχει το 1, την τιμή min = 721 λαμβάνοντας

0 + 721, 29 + 721, 79 + 721, 119 + 721, 129 + 721, 159 + 721, 174 + 721, 179 + 721

ή

721, 750, 800, 840, 850, 880, 895, 900

που είναι τα στοιχεία του αρχικού μας πίνακα ταξινομημένα. Τα στοιχεία αυτά μεταφέρουμε στον αρχικό πίνακα δεδομένων.

Α)Λαμβάνοντας υπόψη την περιγραφή του αλγορίθμου ταξινόμησης, ζητείται να τον υλοποιήσετε με χρήση ψευδοκώδικα, ως ακολούθως:

i) Ορίστε πίνακα δεδομένων μήκους *DATA\_SIZE* και βοηθητικό πίνακα μήκους *RANGE+1* (για ευκολία να θέσετε σταθερές DATA\_SIZE=5 και RANGE=150). Διαβάστε τον πίνακα δεδομένων υποθέτοντας ότι δίνονται διαφορετικοί, μεταξύ τους, μη αρνητικοί αριθμοί (δηλαδή μεγαλύτεροι ή ίσοι με το 0).

ii) Υπολογίστε τον μέγιστο και ελάχιστο αριθμό μεταξύ των δεδομένων που διαβάστηκαν.

iii) Αν ο πίνακας δεδομένων είναι περιορισμένου εύρους για R=150, τότε προχωράτε με τη διαδικασία ταξινόμησης που περιγράφεται στα παραπάνω βήματα 1 και 2, διαφορετικά ενημερώνετε τον χρήστη με κατάλληλο μήνυμα και τερματίζετε το πρόγραμμα.

B)Λαμβάνοντας υπόψη την περιγραφή του αλγορίθμου ταξινόμησης, καθώς και των δεδομένων και ζητούμενων στα Α.i, Α.ii, και Α.iii, ζητείται να τον υλοποιήσετε σε πρόγραμμα με χρήση της γλώσσας προγραμματισμού C.

**ΥΠΟΕΡΓΑΣΙΑ 3. (βαθμοί 35)**

α) Να γραφεί πρόγραμμα σε ψευδοκώδικα, με χρήση υποπρογραμμάτων, το οποίο θα διαχειρίζεται μονοδιάστατους πίνακες Ν ακεραίων αριθμών. Συγκεκριμένα, θα πρέπει να υλοποιηθούν τα παρακάτω υποπρογράμματα:

Υποπρόγραμμα 1: Εισαγωγή στοιχείων στον πίνακα που ορίζει η παράμετρος *Χ*:

***ΔΙΑΔΙΚΑΣΙΑ EIS\_PIN(%X)***

Η διαδικασία αυτή αρχικά ζητά από τον χρήστη **με αμυντικό προγραμματισμό και αντίστοιχο έλεγχο** να δώσει έναν ακέραιο από το 1 έως το 5. Στη συνέχεια εισάγει τον ακέραιο αυτό στην πρώτη θέση του πίνακα, ενώ στη δεύτερη θέση του πίνακα εισάγει τον αμέσως μεγαλύτερό του ακέραιο. Τέλος, στις υπόλοιπες θέσεις του πίνακα εισάγει κάθε φορά το άθροισμα των ακεραίων που υπάρχουν στις δύο προηγούμενες θέσεις του.

Για παράδειγμα, εάν ο χρήστης δώσει στον ζητούμενο ακέραιο την τιμή 1, τότε οι τιμές στον πίνακα που θα εισαχθούν θα είναι 1,2,3,5,8,13,… Σημειώστε ότι η παράμετρος της διαδικασίας αυτής (πίνακας X) είναι παράμετρος κλήσης με αναφορά (*call by reference parameter*), δηλαδή μεταβάλλει την τιμή της μεταβλητής του κυρίως προγράμματος στην οποία αντιστοιχεί.

Υποπρόγραμμα 2: Εμφάνιση στην οθόνη των στοιχείων του πίνακα που ορίζει η παράμετρος *X*:

***ΔΙΑΔΙΚΑΣΙΑ PAROUS\_PIN(X)***

Κατά την εμφάνιση, πριν από κάθε τιμή στοιχείου του πίνακα, να εμφανίζεται η θέση του στοιχείου. Σημειώστε ότι η παράμετρος της διαδικασίας είναι παράμετρος κλήσης με τιμή (*call by value parameter*), δηλαδή δεν μεταβάλλει την τιμή της μεταβλητής του κυρίως προγράμματος στην οποία αντιστοιχεί.

Υποπρόγραμμα 3: Ανάλυση των στοιχείων του πίνακα που ορίζει η παράμετρος *Χ* (κλήση με τιμή) και εισαγωγή στοιχείων στον πίνακα που ορίζει η παράμετρος *Υ* (κλήση με αναφορά).

***ΣΥΝΑΡΤΗΣΗ ANALYSH\_PIN(X, %Y): INTEGER***

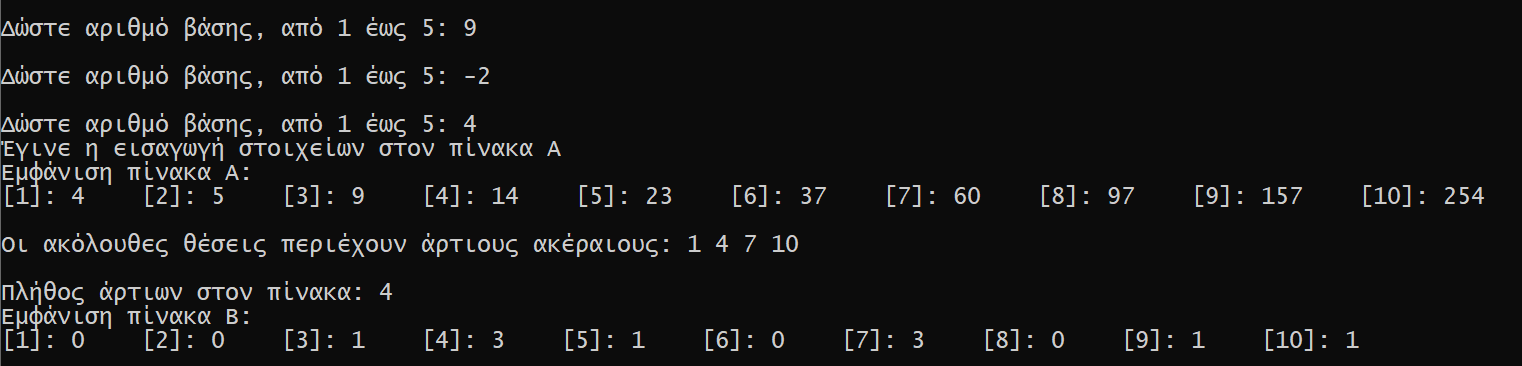
Με τη συνάρτηση αυτή, εκτελούμε δύο λειτουργίες: Αρχικά, διατρέχουμε τον πίνακα X και εμφανίζουμε τις θέσεις που αυτός περιέχει άρτιους ακέραιους. Το πλήθος των άρτιων ακεραίων είναι και η τιμή που επιστρέφει η συνάρτηση. Επιπρόσθετα, η συνάρτηση ελέγχει και τους λήγοντες των ακεραίων του πίνακα X: εξετάζει το πλήθος αυτών που λήγουν σε 1,2,3,…,0 και εισάγει τα πλήθη αυτά στις αντίστοιχες θέσεις του πίνακα *Υ*, δέκα θέσεων*.*

Για παράδειγμα, αν στον πίνακα X οι τέσσερις ακέραιοι λήγουν σε 1, οι τρεις λήγουν σε 4 και οι υπόλοιποι τρεις λήγουν σε 0, τότε όλες οι θέσεις του πίνακα Y θα είχαν τιμή 0, εκτός των Y[1]=4, Y[4]=3 και Y[10]=3.

Στο κυρίως πρόγραμμα, μέσω της κλήσης των παραπάνω υποπρογραμμάτων, να πραγματοποιείται (i) η εισαγωγή στοιχείων σε έναν πίνακα Α (για ευκολία σας να θέσετε τη σταθερά Ν=10 αναφορικά με το μέγιστο πλήθος στοιχείων του), (ii) η εμφάνιση των περιεχομένων του, καθώς και (iii) η ανάλυση των στοιχείων του πίνακα Α και η προβλεπόμενη εισαγωγή τιμών σε ένα δεύτερο πίνακα B, όπως περιγράφεται παραπάνω. Τέλος, να πραγματοποιείται η εμφάνιση του πλήθους των άρτιων καθώς και των θέσεων και των περιεχομένων του πίνακα Β όπως φαίνεται στο ενδεικτικό υπόδειγμα εκτέλεσης.

β) Να υλοποιηθεί το πρόγραμμα του ερωτήματος α, με χρήση της γλώσσας προγραμματισμού C, χωρίς τη χρήση υποπρογραμμάτων (για ευκολία σας να θέσετε τη σταθερά Ν=10 η οποία αναφέρεται στο μέγιστο πλήθος των στοιχείων του πίνακα Α).

Ακολουθεί ενδεικτικό υπόδειγμα εκτέλεσης:



**Γενικές Υποδείξεις:**

Ι) Για τις απαντήσεις της εργασίας μπορείτε να ανατρέξετε στη συμπληρωματική βιβλιογραφία που δίνεται και στα βοηθητικά κείμενα που υπάρχουν στο δικτυακό τόπο / portal της θεματικής ενότητας. Συνιστάται να προσθέσετε στο τέλος της εργασίας σας κατάλογο βιβλιογραφίας.

ΙΙ) Το αρχείο της εργασίας θα πρέπει να έχει ως πρώτη σελίδα το Έντυπο Υποβολής - Αξιολόγησης ΓΕ συμπληρωμένο και ως δεύτερη σελίδα τον τίτλο «Σχόλια προς τον φοιτητή» (θα συμπληρωθεί από τον καθηγητή σας). Οι απαντήσεις στις υπο-εργασίες θα αρχίζουν από την τρίτη σελίδα, χωρίς να επαναλαμβάνονται οι εκφωνήσεις. Κάθε υπο-εργασία θα αρχίζει από νέα σελίδα. Για την απάντησή σας θα πρέπει να χρησιμοποιείτε υποχρεωτικά το **Πρότυπο Υποβολής Γραπτής Εργασίας**.

ΙΙΙ) Οι απαντήσεις πρέπει να είναι γραμμένες με χρήση επεξεργαστή κειμένου (π.χ. Word) σε σελίδες διαστάσεων Α4. Η υποβολή σκαναρισμενών αρχείων και όχι δακτυλογραφημένων δεν είναι αποδεκτή και δεν αξιολογείται. Ενδεικτικά, οι απαντήσεις μπορούν να επιτευχθούν σε περίπου 10 σελίδες.

IV) Για την εφαρμογή του αμυντικού προγραμματισμού αρκεί να γίνεται έλεγχος ως προς το εάν μία τιμή που εισάγεται από το χρήστη του προγράμματος βρίσκεται εντός αποδεκτών ορίων (π.χ. να είναι θετική, διαφορετική από το 0, μεγαλύτερη από 20 κλπ.). Δεν απαιτείται έλεγχος ως προς το εάν η τιμή ανήκει στο σωστό τύπο δεδομένων (π.χ. ακέραιος, πραγματικός αριθμός, χαρακτήρας κλπ.) σύμφωνα με τον τύπο της μεταβλητής στην οποία θα αποθηκευτεί η τιμή αυτή.

V) Στα ερωτήματα όπου ζητείται υλοποίηση κώδικα σε γλώσσα C, στο έγγραφο της απάντησής σας γράψτε μόνο τα τμήματα του προγράμματος που ζητούνται, ενώ στο αρχείο του κώδικα θα πρέπει να δίνεται ολόκληρο το πρόγραμμα. Για να θεωρούνται οι απαντήσεις ολοκληρωμένες θα πρέπει:

α) Ο κώδικας (όπου ζητείται) να είναι επαρκώς σχολιασμένος, σωστά στοιχισμένος και ενσωματωμένος μέσα στο αρχείο με τις απαντήσεις σας σε γραμματοσειρά courier, καθώς και (σύμφωνα με τις οδηγίες του Συμβούλου - Καθηγητή σας) σε ξεχωριστό .c αρχείο (ANSI C).

β) Το όνομα κάθε .c αρχείου να περιλαμβάνει το επώνυμό σας με λατινικούς χαρακτήρες, το χαρακτήρα της υπογράμμισης και τον αριθμό του συγκεκριμένου υποερωτήματος (π.χ. αν το επώνυμό σας είναι Ιωάννου, τότε ο κώδικας για την υποεργασία 1Β θα πρέπει να έχει το όνομα Ioannou\_1b.c).

γ) Κάθε αρχείο .c που θα παραδοθεί θα πρέπει τουλάχιστον να περνά τη φάση της μεταγλώττισης χωρίς λάθη.

VΙ) Η καλή παρουσίαση της εργασίας λαμβάνεται υπόψη στην αξιολόγηση της εργασίας.

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

1. Σύμφωνα με τον Κανονισμό Σπουδών, η καταληκτική ημερομηνία για την παραλαβή της Γ.Ε. από το μέλος ΣΕΠ είναι η επόμενη Τετάρτη από το τέλος της εβδομάδας παράδοσης Γ.Ε. [↑](#footnote-ref-1)