

ΣΕΤ ΑΣΚΗΣΕΩΝ 2**ΕΡΓΑΣΤΗΡΙΟ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ Ι, ΑΚΑΔΗΜΑΪΚΟ ΕΤΟΣ 2013-2014**

Προθεσμία: 9/12/2013, 22:00

Περιεχόμενα

- [Διαβάστε πριν ξεκινήσετε](#)
- [Εκφώνηση άσκησης 1](#)
- [Εκφώνηση άσκησης 2](#)
- [Οδηγίες αποστολής άσκησης](#)

Πριν ξεκινήσετε (ΔΙΑΒΑΣΤΕ ΑΥΤΗ ΤΗΝ ΕΝΟΤΗΤΑ!!)

Διαβάστε ΟΛΗ την εκφώνηση προσεκτικά και “σχεδιάστε” το πρόγραμμά σας στο χαρτί.

Για κάθε στάδιο, αποφασίστε τι μεταβλητές θα χρειαστείτε, τι ονόματα θα τους δώσετε, αν χρειάζονται σταθερές κι αν ναι για ποιες ποσότητες, τι συναρτήσεις θα ορίσετε και τι δομές θα χρησιμοποιήσετε για κάθε λειτουργία.

Μη διστάζετε να ζητήσετε βοήθεια! Μπορείτε να χρησιμοποιήσετε κατά προτίμηση το forum προγραμματισμού (http://courses.inf.uth.gr/coding/?page_id=143) και μόνο αν είναι απαραίτητο email (π.χ. αν πραγματικά επιβάλλεται να στείλετε κάποιο κομμάτι κώδικα μαζί με το μήνυμά σας).

Μην προσπαθήσετε να γράψετε το πρόγραμμα μονομιάς. Για κάθε στάδιο που ολοκληρώνετε, πρέπει να βεβαιώνετε ότι κάνει compile και λειτουργεί σωστά και μετά να προχωράτε στο επόμενο στάδιο. Κάθε φορά που τελειώνετε ένα στάδιο, να αποθηκεύετε ένα αντίγραφο του προγράμματος όπως είναι μέχρι εκείνο το σημείο. Έτσι, αν σε κάποιο επόμενο στάδιο καταστραφεί το αρχείο σας, θα έχετε μια παλιότερη έκδοση από την οποία μπορείτε να συνεχίσετε. Τις επιμέρους “εκδόσεις” να τις ονομάζετε με διαφορετικά ονόματα (πχ. hw2stadiao1.c, hw2stadiao2.c)

Σε κάθε στάδιο σας δίνουμε μια εκτίμηση της δυσκολίας του (διαβάθμιση από 1 έως 3 αστεράκια).

Η εργασία αυτή μπορεί να γίνει σε ομάδες μέχρι 2 ατόμων. Δεν είναι απαραίτητο να συνεργαστείτε με το ίδιο άτομο που κάνατε τα εβδομαδιαία εργαστήρια ή το πρώτο σετ ασκήσεων. Μπορείτε να συζητάτε τις ασκήσεις με συμφοιτητές σας αλλά δεν επιτρέπεται η ανταλλαγή κώδικα με οποιοδήποτε τρόπο.

Ξεκινήστε νωρίς! Ο προγραμματισμός είναι πάντα ΠΟΛΥ πιο χρονοβόρος από ό,τι περιμένετε.

Εκπρόθεσμες ασκήσεις δε γίνονται δεκτές.

Άσκηση 1: Συστάσεις βιβλίων

Εισαγωγή

Κάθε φορά που βλέπετε ένα βίντεο στο youtube, σας συστήνει εναλλακτικά κλιπ για τα οποία πιστεύει ότι θα ενδιαφέρεστε. Ομοίως, online βιβλιοπωλεία όπως το amazon κάνουν συστάσεις βιβλίων με βάση τις προτιμήσεις που έχετε δείξει μέχρι στιγμής.

Η ιδέα πίσω από το μηχανισμό συνιστώμενων βίντεο ή βιβλίων είναι απλή: αν έχετε βαθμολογήσει μια σειρά από αυτά με βάση τις προτιμήσεις σας, και κάποιος άλλος χρήστης έχει παρόμοιες προτιμήσεις (όπως αυτές διαμορφώνονται από τις βαθμολογίες που έδωσε), τότε είναι πολύ πιθανό να σας αρέσουν κάποια βίντεο ή βιβλία που άρεσαν και στον άλλο χρήστη.

Σε αυτό το σετ ασκήσεων θα γράψετε ένα πρόγραμμα το οποίο διαβάζει τις προτιμήσεις ενός μεγάλου αριθμού χρηστών για ένα σύνολο βιβλίων, και τις χρησιμοποιεί για να συστήνει νέα βιβλία σε κάποιον από αυτούς τους χρήστες.

Ακολουθούν λεπτομερείς οδηγίες για το πώς πρέπει να λειτουργεί το πρόγραμμά σας και στάδια κατασκευής του. ΜΗΝ προσπαθήσετε να γράψετε όλο το πρόγραμμα σε ένα βήμα γιατί θα κάνετε λάθη και θα σας πάρει πολύ περισσότερο χρόνο.

Αποθηκεύστε το πρόγραμμα σε αρχείο με όνομα `hw2.c` . Θα σας δώσουμε αρκετά αρχεία τα οποία θα μπορείτε να χρησιμοποιήσετε ως είσοδο (με ανακατεύθυνση) για να ελέγξετε την ορθότητα του προγράμματός σας. Το μεγαλύτερο από αυτά είναι το `data.txt`.

Άσκηση 1, Στάδιο 0: Μελέτη του αρχείου δεδομένων (-)

Ανοίξτε το αρχείο `data.txt` με έναν επεξεργαστή κειμένου. Η δομή του αρχείου είναι ως εξής:

Η πρώτη γραμμή περιέχει έναν ακέραιο αριθμό ο οποίος αναπαριστά το πλήθος των βιβλίων που είναι διαθέσιμα από ένα online βιβλιοπωλείο. Ο αριθμός είναι τουλάχιστον 1 και το πολύ 60.

Ακολουθούν οι περιγραφές τόσων βιβλίων όσος είναι ο παραπάνω αριθμός. Τα βιβλία δίνονται με τη μορφή τίτλος, κόμμα, όνομα συγγραφέα. Οι πληροφορίες για κάθε ένα βιβλίο βρίσκονται όλες σε μια γραμμή και καταλαμβάνουν το πολύ 120 χαρακτήρες. Διαδοχικά βιβλία χωρίζονται από ένα χαρακτήρα αλλαγής γραμμής. Δεν υπάρχουν χαρακτήρες whitespace ανάμεσα στις λέξεις που αποτελούν τον τίτλο ενός βιβλίου ή το όνομα του συγγραφέα.

Ξεκινώντας στη γραμμή μετά τα βιβλία βρίσκονται πληροφορίες για τις βαθμολογίες που έδωσαν οι χρήστες του online βιβλιοπωλείου. Για κάθε χρήστη, το αρχείο περιλαμβάνει:

- το όνομα του χρήστη, το οποίο δεν περιέχει χαρακτήρες whitespace. Τα ονόματα είναι μοναδικά και καταλαμβάνουν το πολύ 20 χαρακτήρες.
- αλλαγή γραμμής
- τους βαθμούς που έδωσε αυτός ο χρήστης, έναν για κάθε ένα από τα παραπάνω βιβλία, με την ίδια σειρά. Οι βαθμοί παίρνουν τις τιμές: -5 (δε μου άρεσε καθόλου), -3 (δε μου άρεσε πολύ), 0 (δεν το έχω διαβάσει), 1 (μέτριο), 3 (μου άρεσε), 5 (μου άρεσε πάρα πολύ). Διαδοχικοί βαθμοί χωρίζονται από whitespace.
- αλλαγή γραμμής.

Υπάρχουν τουλάχιστον ένας και το πολύ 100 χρήστες.

Στη γραμμή μετά τις βαθμολογίες του τελευταίου χρήστη βρίσκεται η λέξη `EndOfNames` η οποία μαρκάρει το τέλος των βαθμολογιών.

Στη γραμμή μετά το `EndOfNames` βρίσκεται το `username` του χρήστη ο οποίος αναζητά προτάσεις για νέα βιβλία. Από εδώ και στο εξής θα τον αποκαλούμε "**αγοραστή**". Ο αγοραστής θα πρέπει να είναι ένας από τους χρήστες, διαφορετικά το πρόγραμμα δε θα μπορεί να δώσει προτάσεις (γιατί δε θα έχει πληροφορίες για τις προτιμήσεις του αγοραστή). Όταν θα ελέγχετε την ορθότητα του προγράμματός σας, καλό είναι να το εκτελέσετε για διάφορους αγοραστές.

Επίσης, επειδή το αρχείο είναι μεγάλο, θα σας δώσουμε και μια μικρότερη έκδοσή του για να διευκολυνθείτε στην αποσφαλμάτωση.

Άσκηση 1, Στάδιο 1: Δημιουργία δομών και εισαγωγή δεδομένων (*)

Διαβάστε από το πληκτρολόγιο μια σειρά από βιβλία και αποθηκεύστε τα ως συμβολοσειρές σε ένα δισδιάστατο πίνακα χαρακτήρων. Κάθε βιβλίο (τίτλος και συγγραφέας) αποθηκεύεται σε μια γραμμή του πίνακα.

Διαβάστε από το πληκτρολόγιο τα ονόματα χρηστών και τις βαθμολογίες που έδωσαν για τα βιβλία. Τα ονόματα πρέπει να τα αποθηκεύσετε ως συμβολοσειρές σε ένα δισδιάστατο πίνακα χαρακτήρων (ένα όνομα χρήστη σε κάθε γραμμή του πίνακα). Τις βαθμολογίες πρέπει να τις αποθηκεύσετε σε ένα δισδιάστατο πίνακα ακεραίων. Εννοείται πως το όνομα ενός χρήστη στη γραμμή *i* του πίνακα ονομάτων αντιστοιχεί στους βαθμούς που καταγράφονται στη γραμμή *i* του πίνακα βαθμολογιών ("παράλληλοι" πίνακες).

Προσωρινός κώδικας: Εκτυπώστε τα περιεχόμενα και των τριών πινάκων και επιβεβαιώστε ότι αυτό το στάδιο λειτουργεί σωστά.

Το πρόγραμμά σας πρέπει να είναι γραμμένο με τέτοιο τρόπο ώστε αν αλλάξει το πλήθος βιβλίων ή ο μέγιστος αριθμός χρηστών, να μπορούν να γίνουν εύκολα και γρήγορα οι κατάλληλες αλλαγές στον κώδικα.

Ολοκληρώστε αυτό το στάδιο και βεβαιωθείτε ότι λειτουργεί σωστά πριν προχωρήσετε στο επόμενο.

Άσκηση 1, Στάδιο 2: Εντοπισμός προτιμήσεων αγοραστή ()**

Αφαιρέστε τον προσωρινό κώδικα που γράψατε στο στάδιο 1.

Εκτυπώστε το μήνυμα: **Hi, what's your name?** ακολουθούμενο από ένα κενό.

Διαβάστε από το πληκτρολόγιο το όνομα του αγοραστή.

Εκτυπώστε χαρακτήρα αλλαγής γραμμής, το μήνυμα **Hello X!** όπου **X** το όνομα που διαβάσατε, και πάλι χαρακτήρα αλλαγής γραμμής.

Χρησιμοποιώντας το όνομα και το γεγονός ότι οι πίνακες ονομάτων/βαθμολογιών είναι "παράλληλοι", βρείτε σε ποια θέση του πίνακα βαθμολογιών βρίσκονται οι βαθμολογίες που έχει δώσει ο αγοραστής.

Αν δε βρεθεί χρήστης με αυτό το όνομα, εκτυπώστε χαρακτήρα αλλαγής γραμμής, το μήνυμα **No such user!** ακολουθούμενο από χαρακτήρα αλλαγής γραμμής κι επαναλάβετε τη διαδικασία αυτού του σταδίου.

Αν βρεθεί, εκτυπώστε χαρακτήρα αλλαγής γραμμής, το μήνυμα **User #Y** όπου **Y** η θέση που βρήκατε, και πάλι χαρακτήρα αλλαγής γραμμής.

Ολοκληρώστε αυτό το στάδιο και βεβαιωθείτε ότι λειτουργεί σωστά πριν προχωρήσετε στο επόμενο.

Άσκηση 1, Στάδιο 3: Εύρεση χρήστη με τις πιο όμοιες προτιμήσεις (*)**

Για κάθε έναν από τους υπόλοιπους χρήστες, υπολογίστε το βαθμό ομοιότητας των βαθμολογιών τους με τις βαθμολογίες του αγοραστή, και βρείτε το χρήστη με το μεγαλύτερο βαθμό ομοιότητας. Σε περίπτωση ισοβαθμίας, επιλέξτε όποιο χρήστη βρέθηκε πρώτος.

Για να υπολογίσουμε το βαθμό ομοιότητας των βαθμολογιών δύο χρηστών αντιμετωπίζουμε τις βαθμολογίες που έδωσαν ως δύο διανύσματα και υπολογίζουμε το εσωτερικό τους γινόμενο.

Εκτυπώστε **χαρακτήρα αλλαγής γραμμής**, **το όνομα του χρήστη που βρήκατε**, **ένα κενό**, **το βαθμό ομοιότητας** **κι ένα χαρακτήρα αλλαγής γραμμής**. (Αυτό κανονικά δε θα έπρεπε να δημοσιοποιείται, αλλά το εκτυπώνετε για να είναι πιο εύκολος ο έλεγχος των αποτελεσμάτων σας)

Ολοκληρώστε αυτό το στάδιο και βεβαιωθείτε ότι λειτουργεί σωστά πριν προχωρήσετε στο επόμενο.

Άσκηση 1, Στάδιο 4: Εύρεση συνιστώμενου βιβλίου ()**

Ψάξτε στα βιβλία του χρήστη που έχει τις πιο όμοιες προτιμήσεις με τον αγοραστή και εντοπίστε το πρώτο βιβλίο που του άρεσε (δηλαδή έχει βαθμό > 0) και που δεν το έχει διαβάσει ακόμη ο αγοραστής.

Εκτυπώστε χαρακτήρα αλλαγής γραμμής, το μήνυμα **We recommend: X** όπου **X** τα στοιχεία του βιβλίου (τίτλος, συγγραφέας) και χαρακτήρα αλλαγής γραμμής.

Αν δε βρεθεί κατάλληλο βιβλίο, εκτυπώστε χαρακτήρα αλλαγής γραμμής, το μήνυμα **Sorry, no recommendations!** και χαρακτήρα αλλαγής γραμμής.

Άσκηση 1, Στάδιο 5: Τελικός έλεγχος ορθότητας

Το πρόγραμμά σας πρέπει να λειτουργεί σωστά και να εκτυπώνει όλα τα μηνύματα και αποτελέσματα με τον τρόπο που σας περιγράφουμε. Θα σας δώσουμε ενδεικτικά αρχεία εισόδου και εξόδου. Χρησιμοποιήστε ανακατεύθυνση εισόδου/εξόδου και την εντολή **diff** όπως κάνατε και στο πρώτο σετ ασκήσεων για να επιβεβαιώσετε ότι το πρόγραμμά σας βγάζει ακριβώς την αναμενόμενη έξοδο.

Πρέπει το πρόγραμμα που θα μας παραδώσετε να παράγει έξοδο που δεν έχει διαφορές από τη δική μας.

Αρχείο προς παράδοση: hw2.c

Άσκηση 2: Συστάσεις βιβλίων, το sequel

Εισαγωγή

Στη δεύτερη άσκηση θα εκμεταλλευτείτε το γεγονός ότι κάθε ένα από τα στάδια που ολοκληρώσατε στην άσκηση 1 είναι μια καλά ορισμένη λειτουργία η οποία μπορεί να υλοποιηθεί ως ξεχωριστή συνάρτηση, και θα ξαναγράψετε το πρόγραμμά σας ώστε να χρησιμοποιεί συναρτήσεις.

Για να σας διευκολύνουμε, θα επιτρέψουμε περιορισμένη χρήση καθολικών μεταβλητών. Για την ακρίβεια, οι τρεις πίνακες στους οποίους αποθηκεύονται τα βιβλία, τα ονόματα χρηστών και οι βαθμολογίες επιτρέπεται να δηλωθούν καθολικά στο πρόγραμμα, το οποίο σημαίνει ότι δε χρειάζεται να περαστούν ως παράμετροι στις συναρτήσεις. Όλες οι άλλες μεταβλητές του προγράμματος πρέπει να δηλωθούν τοπικά.

Το τελικό σας πρόγραμμα πρέπει να έχει την ίδια συμπεριφορά με το πρόγραμμα που γράψατε για την άσκηση 1, εκτός από το στάδιο 4.

Αποθηκεύστε το πρόγραμμα σε αρχείο με το όνομα `hw2functions.c`.

Άσκηση 2, Στάδιο 1: Δημιουργία δομών και εισαγωγή δεδομένων (*)

Γράψτε μια συνάρτηση η οποία δεν παίρνει παραμέτρους, διαβάζει από το πληκτρολόγιο το πλήθος των βιβλίων καθώς και τα χαρακτηριστικά τους (τίτλος, συγγραφέας) και τα αποθηκεύει στον αντίστοιχο πίνακα. Η συνάρτηση επιστρέφει το πλήθος των βιβλίων που διαβάστηκαν.

Γράψτε μια συνάρτηση η οποία παίρνει ως παράμετρο το πλήθος των βιβλίων, διαβάζει από το πληκτρολόγιο τα ονόματα και τις βαθμολογίες των χρηστών, τα αποθηκεύει στους αντίστοιχους πίνακες, κι επιστρέφει το πλήθος χρηστών που έχουν βαθμολογήσει βιβλία.

Χρησιμοποιήστε τις συναρτήσεις που γράψατε για να διαβάσετε τα δεδομένα του προγράμματος και να τα αποθηκεύσετε στους κατάλληλους πίνακες.

Ολοκληρώστε αυτό το στάδιο και βεβαιωθείτε ότι λειτουργεί σωστά πριν προχωρήσετε στο επόμενο.

Άσκηση 2, Στάδιο 2: Εντοπισμός προτιμήσεων αγοραστή (*)

Γράψτε μια συνάρτηση η οποία παίρνει ως παράμετρο το πλήθος χρηστών, διαβάζει το όνομα ενός αγοραστή όπως περιγράφεται στο στάδιο 2 της άσκησης 1, εντοπίζει τη θέση στον πίνακα βαθμολογιών όπου βρίσκονται οι βαθμολογίες του αγοραστή, και την επιστρέφει. Αν δε βρεθεί χρήστης με αυτό το όνομα, εκτυπώνει μήνυμα όπως περιγράφεται στο στάδιο 2 της άσκησης 1, κι επιστρέφει -1.

Χρησιμοποιήστε αυτή τη συνάρτηση, με την κλήση της μέσα σε επανάληψη μέχρις ότου δοθεί όνομα αγοραστή που να υπάρχει στον πίνακα χρηστών.

Ολοκληρώστε αυτό το στάδιο και βεβαιωθείτε ότι λειτουργεί σωστά πριν προχωρήσετε στο επόμενο.

Άσκηση 2, Στάδιο 3: Εύρεση χρήστη με τις πιο όμοιες προτιμήσεις (*)

Γράψτε μια συνάρτηση η οποία παίρνει ως παράμετρο το πλήθος χρηστών, τη θέση όπου βρίσκεται ο αγοραστής και το πλήθος βιβλίων, κι επιστρέφει τη θέση στον πίνακα χρηστών όπου βρίσκεται ο χρήστης με το μεγαλύτερο βαθμό ομοιότητας ανάμεσα στις προτιμήσεις του και τις προτιμήσεις του αγοραστή.

Αφού χρησιμοποιήσετε τη συνάρτηση για να βρείτε το χρήστη με το μεγαλύτερο βαθμό ομοιότητας, εκτυπώστε χαρακτήρα αλλαγής γραμμής, το όνομά του, ένα κενό, το βαθμό ομοιότητας κι ένα χαρακτήρα αλλαγής γραμμής, ακριβώς όπως περιγράφεται στο στάδιο 3 της άσκησης 1.

Ολοκληρώστε αυτό το στάδιο και βεβαιωθείτε ότι λειτουργεί σωστά πριν προχωρήσετε στο επόμενο.

Άσκηση 2, Στάδιο 4: Εύρεση συνιστώμενων βιβλίων ()**

Γράψτε μια συνάρτηση η οποία παίρνει ως παραμέτρους τη θέση στον πίνακα ονομάτων όπου βρίσκεται το όνομα του αγοραστή, τη θέση όπου βρίσκεται το όνομα του χρήστη με τις πιο όμοιες προτιμήσεις, και το πλήθος βιβλίων, και κάνει τα παρακάτω:

- κατασκευάζει ένα διδιάστατο πίνακα συμβολοσειρών στον οποίο θα αποθηκεύσει τα στοιχεία (τίτλος, συγγραφέας) όλων των βιβλίων που θα προταθούν στον αγοραστή.
- γεμίζει τον πίνακα με όλα τα βιβλία που έχει βαθμολογήσει θετικά ο χρήστης με τις όμοιες προτιμήσεις και τα οποία δεν έχει ήδη διαβάσει ο αγοραστής.
- Εκτυπώνει χαρακτήρα αλλαγής γραμμής, το μήνυμα **We recommend:** ακολουθούμενο από χαρακτήρα αλλαγής γραμμής, και μετά τους τίτλους και συγγραφείς όλων των βιβλίων που βρίσκονται στον πίνακα. Τα στοιχεία κάθε βιβλίου εμφανίζονται σε νέα σειρά και στην αρχή της κάθε σειράς υπάρχει χαρακτήρας tab. Αν ο πίνακας δεν περιέχει κανένα βιβλίο, τότε εκτυπώνεται μόνο χαρακτήρας αλλαγής γραμμής και το μήνυμα **Sorry, no recommendations!** ακολουθούμενο από χαρακτήρα αλλαγής γραμμής.

Άσκηση 2, Στάδιο 5: Τελικός έλεγχος ορθότητας

Το πρόγραμμά σας πρέπει να λειτουργεί σωστά και να εκτυπώνει όλα τα μηνύματα και αποτελέσματα με τον τρόπο που σας περιγράφουμε. Θα σας δώσουμε ενδεικτικά αρχεία εισόδου και εξόδου. Χρησιμοποιήστε ανακατεύθυνση εισόδου/εξόδου και την εντολή diff όπως κάνατε και στο πρώτο σετ ασκήσεων για να επιβεβαιώσετε ότι το πρόγραμμά σας βγάζει ακριβώς την αναμενόμενη έξοδο.

Πρέπει το πρόγραμμα που θα μας παραδώσετε να παράγει έξοδο που δεν έχει διαφορές από τη δική μας.

Αρχείο προς παράδοση: hw2fun.c

Παρατήρηση σχετικά με τις προτάσεις βιβλίων: Ίσως παρατηρήσετε ότι η μέθοδος για τον υπολογισμό ομοιότητας ανάμεσα στις βαθμολογίες δύο χρηστών δεν είναι πάντα πολύ ακριβής. Παρόλα αυτά, την προτιμήσαμε γιατί είναι απλή και στις πιο πολλές περιπτώσεις δίνει καλά αποτελέσματα.

Πώς να παραδώσετε τη δουλειά σας

(Ακολουθείστε τις οδηγίες ακριβώς αλλιώς μπορεί να μη δούμε τα αρχεία σας)

Προσθέστε σε σχόλια στην αρχή του κάθε αρχείου με κώδικα τα πλήρη ονόματα και ΑΜ των μελών της ομάδας. Παρακαλούμε να γράφετε τα σχόλια ΜΟΝΟ με λατινικούς χαρακτήρες.

Κατασκευάστε ένα φάκελο με όνομα `hw2_epwnumero1_AM1_epwnumero2_AM2` και αντιγράψτε μέσα σε αυτόν το `hw2.c` και το `hw2fun.c`

Πηγαίνετε στο φάκελο μέσα στον οποίο βρίσκεται το `epwnumero1_AM1_epwnumero2_AM2` που κατασκευάσατε και γράψτε την παρακάτω εντολή:

```
tar czvf hw2_epwnumero1_AM1_epwnumero2_AM2.tar.gz hw2_epwnumero1_AM1_epwnumero2_AM2
```

Στείλτε email:

- στη διεύθυνση **`ce120lab@gmail.com`**
- αντίγραφο (CC) στον άλλο μέλος της ομάδας σας
- θέμα (subject) **CE120 hw2**
- και επικολλημένο αρχείο το `hw2_epwnumero1_AM1_epwnumero2_AM2.tar.gz`

Το πρόγραμμά σας θα βαθμολογηθεί ως προς:

- ορθότητα υπολογισμών
- σωστή επιλογή τύπων
- σωστή επιλογή και χρήση δομών ελέγχου κι επανάληψης
- σωστή χρήση πινάκων
- σωστός ορισμός και χρήση συναρτήσεων
- αναγνωσιμότητα (ονόματα, στοίχιση, σχολιασμός, μορφή κώδικα)
- φορμαρισμένη είσοδος/έξοδος και συμμόρφωση με τις προδιαγραφές.