

ΣΕΤ ΑΣΚΗΣΕΩΝ 3

ΕΡΓΑΣΤΗΡΙΟ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ Ι, ΑΚΑΔΗΜΑΪΚΟ ΕΤΟΣ 2016-2017

Προθεσμία: Τετάρτη 10/12/2017, 22:00

Διαβάστε πριν ξεκινήσετε

Διαβάστε όλη την εκφώνηση προσεκτικά πριν ξεκινήσετε να γράφετε κώδικα. Ολοκληρώστε κάθε ένα στάδιο πριν προχωρήσετε στο επόμενο, κι αποθηκεύστε σε ξεχωριστά αρχεία ενδιαμέσες σωστές μορφές του προγράμματός σας ώστε να μπορείτε να επανέλθετε σε αυτές αν κάνετε κάποιο λάθος.

Η εργασία αυτή μπορεί να γίνει σε ομάδες **μέχρι 2 ατόμων**. Δε χρειάζεται να είστε ομάδα με το ίδιο άτομο που είστε στο εργαστήριο. Μπορείτε να συζητάτε τις ασκήσεις με συμφοιτητές σας αλλά δεν επιτρέπεται η ανταλλαγή κώδικα με οποιονδήποτε τρόπο.

Ξεκινήστε νωρίς! Ο προγραμματισμός είναι πάντα ΠΟΛΥ πιο χρονοβόρος από ότι περιμένετε.

Εκπρόθεσμες ασκήσεις δε γίνονται δεκτές.

Οι ασκήσεις σας θα βαθμολογηθούν στα παρακάτω (χωρίς ιδιαίτερη σειρά):

- Ορθότητα
- Αναγνωσιμότητα (στοίχιση, ονόματα μεταβλητών, κτλ.)
- Σχεδιασμός προγράμματος και αποτελεσματική χρήση κατάλληλων δομών, μεταβλητών, σταθερών, συναρτήσεων κτλ.
 - Ειδικά για τις συναρτήσεις, θα πρέπει να αποφασίσετε εσείς πώς να οργανώσετε τον κώδικά σας και τι θα ήταν καλό να γίνει σε συνάρτηση. Δε θα δώσουμε ιδιαίτερες οδηγίες για κάτι τέτοιο.
- Συμμόρφωση με τις προδιαγραφές
- Αποτελεσματικά σχόλια, σύμφωνα με τους κανόνες σχολιασμού του σχετικού φυλλαδίου.

Προσέξτε ιδιαίτερα:

- Τα όρια των loops όταν κάνετε προσπέλαση πινάκων.
- Μην επαναλαμβάνετε μεγάλα κομμάτια κώδικα (π.χ. αν έχετε ίδιες εντολές και στα δύο σκέλη μιας if/else, βγάλτε τις απέξω), ή χρησιμοποιήστε συνάρτηση.
- **Απαγορεύεται να χρησιμοποιήσετε καθολικές/static μεταβλητές, goto, gets και ότι άλλο έχουμε "απαγορεύσει" στις εργασίες.**

Άσκηση 1: Επεξεργασία εικόνας

Στην παρούσα άσκηση θα γράψετε ένα πρόγραμμα επεξεργασίας εικόνας. Μπορείτε να φανταστείτε μία εικόνα σαν ένα διδιάστατο πίνακα από εικονοστοιχεία (στην αγγλική βιβλιογραφία αναφέρονται ως *pixels*), μεγέθους όσο και το μέγεθος της εικόνας. Για παράδειγμα, μία εικόνα 712x512 εικονοστοιχεία αντιστοιχεί σε ένα διδιάστατο πίνακα από εικονοστοιχεία 712 στηλών και 512 γραμμών (συνηθίζουμε να αναφέρουμε πρώτα το πλάτος και μετά το ύψος της εικόνας).

Κάθε εικονοστοιχείο απεικονίζει τις τιμές φωτεινότητας για τα τρία βασικά χρώματα κόκκινο, πράσινο και μπλε, γνωστά και ως RGB από τα αρχικά των λέξεων *red*, *green*, *blue* στην αγγλική γλώσσα. Θεωρούμε ότι οι τιμές της φωτεινότητας είναι πάντα φυσικοί αριθμοί στο εύρος $[0, 255]$.

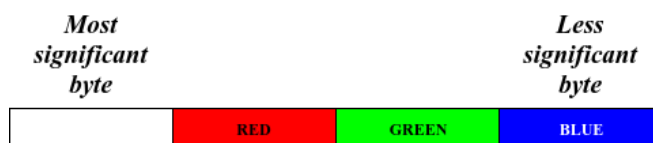
Παραδείγματα **RGB** τιμών είναι τα εξής:

1. Ένα **κόκκινο** pixel έχει RGB τιμές 255, 0, 0 (μέγιστη τιμή φωτεινότητας για το κόκκινο και μηδέν για τα υπόλοιπα).
2. Ένα **κίτρινο** pixel προκύπτει από τις RGB τιμές 255, 255, 0 (το κίτρινο προκύπτει από τη μίξη του κόκκινου με το πράσινο χρώμα)
3. Ένα άσπρο pixel προκύπτει από τις RGB τιμές 255, 255, 255 (μέγιστη φωτεινότητα για όλα τα χρώματα)
4. Ένα **μαύρο** pixel προκύπτει από τις RGB τιμές 0, 0, 0 (ελάχιστη φωτεινότητα για όλα τα χρώματα)

Μπορείτε να δοκιμάσετε διαφορετικούς χρωματικούς συνδυασμούς [εδώ](#) ή με χρήση κάποιου προγράμματος επεξεργασίας εικόνας (π.χ. [gimp](#)).

Εφόσον οι τιμές των εικονοστοιχείων για κάθε βασικό RGB χρώμα κυμαίνονται μεταξύ 0 και 255, οποιαδήποτε τιμή χωράει σε 1 byte (αντιστοιχούν στις 2^8 διαφορετικές διατάξεις των 8bit). Για κάθε εικονοστοιχείο διατηρούμε τρία χρώματα, οπότε χρειαζόμαστε συνολικά 3 bytes για την αποθήκευση του. Εναλλακτικά, οι τιμές και των τριών χρωμάτων θα μπορούσαν να αποθηκευτούν σε ένα ακέραιο μεγέθους 4 byte, όπου κάθε χρώμα αποθηκεύεται σε ένα από τα διαφορετικά bytes του ακεραίου. Η παρούσα άσκηση υιοθετεί αυτή την προσέγγιση.

Το παρακάτω σχήμα παρουσιάζει μία από τις πιθανές προτάσεις αποθήκευσης των RGB χρωμάτων σε έναν ακέραιο μεγέθους 4 bytes. Για την αποθήκευση στον ακέραιο και την ανάκτηση της πληροφορίας από τον ακέραιο συνιστάται να χρησιμοποιήσετε δεξιά και αριστερή ολίσθηση (*bit-shift*), όπου απαιτείται.



Σχήμα 1: Αποθήκευση των τριών χρωμάτων RGB σε ακέραιο μεγέθους 4 bytes

Η μορφή της εικόνας PPM

Σημείωση: Το διάβασμα από αρχείο τύπου PPM σας δίνεται έτοιμο, με εξαίρεση την αποθήκευση της πληροφορίας των τριών χρωμάτων ενός εικονοστοιχείου σε έναν ακέραιο. Η αποθήκευση της πληροφορίας του εικονοστοιχείου σε ακέραιο θα γίνει από εσάς μέσω κατάλληλης συνάρτησης.

Στην παρούσα εργασία θα χρησιμοποιήσετε για διάβασμα από αρχείο και αποθήκευση σε αρχείο εικόνες που είναι αποθηκευμένες σε μορφή (format) PPM. Πρόκειται για έγχρωμες εικόνες που η πληροφορία της εικόνας αποθηκεύεται σε μορφή κειμένου. Η μορφή ενός αρχείου PPM έχει ως εξής:

1. Ξεκινάει με το αλφαριθμητικό **P3**.

2. Ακολουθεί ένας ακέραιος που αντιστοιχεί στο πλάτος της εικόνας (σε εικονοστοιχεία).
3. Ακολουθεί ένας ακέραιος που αντιστοιχεί στο ύψος της εικόνας (σε εικονοστοιχεία).
4. Ακολουθεί ένας ακέραιος που αντιστοιχεί στη μέγιστη τιμή φωτεινότητας της εικόνας (για τις δικές μας εικόνες η μέγιστη τιμή φωτεινότητας θα είναι 255).

Τα σημεία 1-4 αποτελούν την κεφαλίδα του αρχείου.

5. Στη συνέχεια για κάθε εικονοστοιχείο εμφανίζονται 3 ακέραιοι για τα τρία **RGB** χρώματα κόκκινο, πράσινο και μπλε με τη σειρά. Το αρχείο ξεκινά με την πληροφορία του επάνω αριστερού εικονοστοιχείου (θεωρούμε ως πρώτη σειρά εικονοστοιχείων, την κορυφαία σειρά), έπεται το αμέσως δεξιότερο εικονοστοιχείο, μέχρι να φτάσουμε στο δεξιότερο στοιχείο της κορυφαίας σειράς. Στη συνέχεια το αρχείο συνεχίζει από τον αριστερότερο στοιχείο της επόμενης σειράς κ.ο.κ.

Για παράδειγμα, εάν έχουμε μία εικόνα μεγέθους **25x25** εικονοστοιχείων στο αρχείο θα έχουμε αποθηκευμένους **25x25x3** ακεραίους που αντιστοιχούν στην πληροφορία των εικονοστοιχείων της εικόνας. Είναι προφανές ότι η τιμή κάθε ακεραίου δεν πρέπει να υπερβαίνει τη μέγιστη τιμή φωτεινότητας που ορίστηκε στην αρχή του αρχείου (στο σημείο 4).

Οποιοδήποτε αλφαριθμητικό εντός του αρχείου διαχωρίζεται από την υπόλοιπη πληροφορία με έναν ή περισσότερους κενούς χαρακτήρες, χαρακτήρες tab ή χαρακτήρες αλλαγής γραμμής ή συνδυασμούς των παραπάνω.

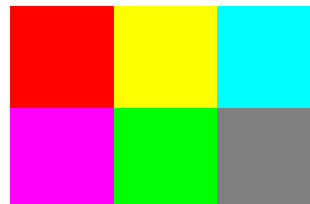
Παρακάτω δίνεται ένα παράδειγμα εικόνας PPM μεγέθους 2x3 και δίπλα η εικόνα που αντιστοιχεί σε αυτό σε μεγέθυνση. Το παρακάτω περιεχόμενο αντιστοιχεί στο αρχείο κειμένου **test.ppm** που σας παρέχεται έτοιμο για τον έλεγχο του προγράμματος σας.

P3

3 2

255

255	0	0	255	255	0	0	255	255
255	0	255	0	255	0	128	128	128



Διάβασμα της εικόνας και αποθήκευση σε πίνακα

Κατ' αρχήν απαραίτητη προϋπόθεση για το διάβασμα της πληροφορίας σε πίνακα είναι ο πίνακας να είναι ικανού μεγέθους, ώστε να μπορεί να αποθηκεύσει την εικόνα. Ο πίνακας μπορεί να έχει διαστάσεις μεγαλύτερες ή ίσες της εικόνας που πρόκειται να αποθηκεύσει, όχι όμως μικρότερες.

Εφόσον ισχύει η παραπάνω προϋπόθεση, το διάβασμα μιας εικόνας και η αποθήκευση σε διδιάστατο πίνακα (δες τη συνάρτηση **load_image**) γίνεται ως εξής. Η πρώτη σειρά εικονοστοιχείων όπως είναι αποθηκευμένη στο αρχείο αποθηκεύεται στην πρώτη γραμμή του πίνακα με την σειρά που αυτά διαβάζονται, η δεύτερη σειρά εικονοστοιχείων στη δεύτερη γραμμή του πίνακα κ.ο.κ.

Η αποθήκευση της εικόνας από τον πίνακα σε αρχείο γίνεται ως εξής. Αρχικά γράφεται η επικεφαλίδα της εικόνας, δηλαδή η συμβολοσειρά P3, το πλάτος, το ύψος και η μέγιστη τιμή για την φωτεινότητα. Στη συνέχεια καταγράφονται οι τιμές RGB όλων των εικονοστοιχείων ανά γραμμή ξεκινώντας από το 1ο στοιχείο της 1η γραμμής του πίνακα.

Περιγραφή του προγράμματος

Για να ελέγξετε το πρόγραμμα σας δίνονται οι εξής εικόνες:

1. **test.ppm** μεγέθους 2x3. Στοιχειώδης εικόνα για την εύκολη αποσφαλμάτωση του κώδικα σας.
2. **lena.ppm** μεγέθους 512x512.
3. **landscape.ppm** μεγέθους 500x314.

Όλες τις εικόνες PPM μπορείτε να τις ανοίξετε για να δείτε το περιεχόμενο τους με χρήση του προγράμματος kate ή οποιουδήποτε άλλου text editor.

Βήμα 1ο – Δήλωση μεταβλητών

Στη συνάρτηση **main** καλείστε να δηλώσετε τις εξής μεταβλητές:

1. ένα διδιάστατο πίνακα ακεραίων μεγέθους **IMG_SIZExIMG_SIZE** (προτείνεται **IMG_SIZE=1024**).
2. Έναν ακέραιο που αποθηκεύει το πλάτος της εικόνας σε εικονοστοιχεία.
3. Έναν ακέραιο που αποθηκεύει το ύψος της εικόνας σε εικονοστοιχεία.
4. Έναν ακέραιο που αποθηκεύει τη μέγιστη τιμή φωτεινότητας για την εικόνα

Έλεγχος: Στη συνάρτηση **main** δεσμεύστε ένα επιπλέον διδιάστατο πίνακα ακεραίων μεγέθους **IMG_SIZExIMG_SIZE**. Εάν το πρόγραμμα σας τερματίζεται εκτυπώνοντας **segmentation fault** μειώστε την τιμή του **IMG_SIZE**, σε τιμή που να μη λαμβάνετε segmentation fault και ενημερώστε μας σχετικά μέσω e-mail.

Βήμα 2ο – Διάβασμα της εικόνας από αρχείο

Γράψτε μία συνάρτηση η οποία λαμβάνει ως παραμέτρους τρεις ακέραιους αριθμούς. Οι τρεις πρώτοι ακέραιοι αντιπροσωπεύουν τις 3 τιμές RGB ενός εικονοστοιχείου. Η συνάρτηση υλοποιεί την αποθήκευση των τριών χρωμάτων RGB του εικονοστοιχείου σε έναν ακέραιο αριθμό κατά τα πρότυπα του σχήματος 1. Η συνάρτηση επιστρέφει έναν ακέραιο που αντιπροσωπεύει την ακέραια τιμή που κατασκευάστηκε.

Σας δίνεται έτοιμη η συνάρτηση **load_image** η οποία λαμβάνει ως παραμέτρους ένα δείκτη σε χαρακτήρα που δείχνει στην αρχή της συμβολοσειράς που περιέχει το file path από το οποίο θα διαβάσουμε την εικόνα, ένα διδιάστατο πίνακα ακεραίων, και 3 δείκτες σε ακέραιο. Μετά το διάβασμα της εικόνας, ο πίνακας θα έχει ενημερωθεί με την πληροφορία των εικονοστοιχείων της εικόνας και οι ακέραιοι στους οποίους δείχνουν οι 3 δείκτες θα περιέχουν ενημερωμένες τιμές αναφορικά με το πλάτος, το ύψος και τη μέγιστη τιμή φωτεινότητας για την εικόνα. Η συνάρτηση επιστρέφει 0 σε περίπτωση επιτυχίας ή -1 σε περίπτωση αποτυχίας.

Στη συνάρτηση **load_image** που σας δίνεται έτοιμη, στο σημείο που υπάρχουν τα σχόλια “**//TODO call function here.**”, καλείστε να καλέσετε τη συνάρτηση που γράψατε παραπάνω, ώστε να αποθηκευθούν οι τιμές των τριών RGB χρωμάτων που διαβάστηκαν μέσω της συνάρτησης **fscanf** (ανάλογη της **scanf** για διάβασμα από αρχείο) στη γραμμή **row** και στη στήλη **col** του διδιάστατου πίνακα ακεραίων στον οποίο αποθηκεύεται τελικά η εικόνα.

Στη συνάρτηση **main** γράψτε κώδικα, ώστε όταν το πρόγραμμα ξεκινά να εμφανίζει το μήνυμα “**Enter file path to load image:** ”. Στη συνέχεια περιμένει από το χρήστη να εισάγει ένα έγκυρο file path από το οποίο θα διαβάσει μία εικόνα. Το file path που θα δώσει ο χρήστης μπορεί να

είναι απόλυτο (π.χ. /home/myusername/images/test.ppm) ή σχετικό ως προς την θέση που βρίσκεται (π.χ. images/test.ppm), υποθέτοντας ότι βρίσκεστε μέσα στον κατάλογο /home/myusername.

Αφού διαβάσει το file path που δίνει ο χρήστης επιχειρεί να διαβάσει την εικόνα από το αρχείο με τη βοήθεια της συνάρτησης **load_image**, που ολοκληρώσατε παραπάνω. Εάν αποτύχει εκτυπώνει το μήνυμα **"Invalid path!"** ακολουθούμενο από χαρακτήρα αλλαγής γραμμής και επαναλαμβάνει τη διαδικασία μέχρι ο χρήστης να δώσει ένα έγκυρο path. Μετά το επιτυχημένο διάβασμα της εικόνας εκτυπώνει το μήνυμα **"Load OK!"** ακολουθούμενο από χαρακτήρα αλλαγής γραμμής.

Βήμα 3ο – Εκτύπωση του περιεχομένου της εικόνας στην οθόνη

Γράψτε μία συνάρτηση η οποία εκτυπώνει τα περιεχόμενα της εικόνας στην οθόνη ως εξής:

- Εκτυπώνει το μήνυμα **"##### IMAGE #####"** (11 χαρακτήρες # σε κάθε πλευρά) ακολουθούμενο από χαρακτήρα αλλαγής γραμμής.
- Εκτυπώνει το πλάτος, το ύψος και τη μέγιστη τιμή φωτεινότητας χωρισμένα μεταξύ τους με τον κενό χαρακτήρα. Ακολουθεί χαρακτήρας αλλαγής γραμμής.
- Εκτυπώνει ανά γραμμή και ξεκινώντας από την 1η γραμμή, όλες τις τιμές του πίνακα των εικονοστοιχείων σε δεκαεξαδική μορφή, έχουν το πρόθεμα **0x**, πλάτος 8 χαρακτήρες και μηδενικά στην αρχή (χρησιμοποιήστε κατάλληλο προσδιοριστή στην printf). Οι τιμές χωρίζονται μεταξύ τους με τον κενό χαρακτήρα. Μετά την εκτύπωση της τελευταίας τιμής κάθε γραμμής ακολουθεί χαρακτήρας αλλαγής γραμμής.
- Εκτυπώνει το μήνυμα **"#####"** (29 χαρακτήρες #) ακολουθούμενο από δύο χαρακτήρες αλλαγής γραμμής.

Στη συνάρτηση **main** καλέστε τη συνάρτηση που μόλις γράψατε, ώστε αφού διαβάσετε την εικόνα **test.ppm** να βεβαιωθείτε ότι τη διαβάζετε σωστά. Εφόσον διαβάζετε σωστά την εικόνα βάλτε σε σχόλια τη συγκεκριμένη γραμμή πριν προχωρήσετε στο επόμενο βήμα.

Βήμα 4ο – Εκτύπωση της εικόνας σε αρχείο

Γράψτε μία συνάρτηση η οποία λαμβάνει ως παραμέτρους έναν ακέραιο και τρεις δείκτες σε ακέραιο. Οι τρεις δείκτες σε ακέραιο δείχνουν σε 3 διαφορετικούς ακέραιους που αντιπροσωπεύουν τις τιμές RGB ενός εικονοστοιχείου. Η συνάρτηση υλοποιεί την εξαγωγή των τριών χρωμάτων RGB του εικονοστοιχείου από τον ακέραιο (1η παράμετρος) στους ακέραιους που δείχνουν οι 3 δείκτες κατά τα πρότυπα του σχήματος 1.

Σας δίνεται έτοιμη η συνάρτηση **save_image** η οποία λαμβάνει ως παραμέτρους ένα δείκτη σε χαρακτήρα που δείχνει στην αρχή της συμβολοσειράς που περιέχει το file path στο οποίο θα γράψουμε την εικόνα, ένα διδιάστατο πίνακα ακεραίων, και 3 ακέραιους που αντιπροσωπεύουν το πλάτος, το ύψος και τη μέγιστη τιμή φωτεινότητας της εικόνας. Η συνάρτηση εκτυπώνει το περιεχόμενο της εικόνας στο αρχείο που προσδιορίζεται από το file path που παρέχεται ως πρώτο όρισμα. Η συνάρτηση επιστρέφει 0 σε περίπτωση επιτυχίας ή -1 σε περίπτωση αποτυχίας.

Στη συνάρτηση **save_image** που σας δίνεται έτοιμη, στο σημείο που υπάρχουν τα σχόλια **"//TODO call function here."**, καλείστε να καλέσετε τη συνάρτηση που γράψατε παραπάνω, ώστε να εξαχθούν οι τιμές των τριών RGB χρωμάτων που είναι αποθηκευμένες στη γραμμή row και τη στήλη col του πίνακα ακεραίων και να αποθηκευθούν στους ακέραιους στους οποίους δείχνουν οι τρεις δείκτες που περνιούνται ως παράμετροι στη συνάρτηση.

Στη συνάρτηση **main**, αφού διαβάσετε με επιτυχία την εικόνα, το πρόγραμμα σας θα πρέπει να εκτυπώνει το παρακάτω μήνυμα, ακολουθούμενο από χαρακτήρα αλλαγής γραμμής. Το μήνυμα

αντιπροσωπεύει το κεντρικό μενού του προγράμματος και εκτυπώνεται αφού το πρόγραμμα διαβάσει την αρχική εικόνα (βήμα 2) και κάθε φορά που ολοκληρώνεται μία εντολή και ο χρήστης καλείται να εισάγει την επόμενη.

Enter option:

Print image to screen: (P/p)

Load another image: (L/l)

Save image: (S/s)

Quit: (Q/q)

Οι επιλογές τις οποίες δίνονται στον χρήστη είναι οι εξής:

- Εισάγοντας **P** ή **p** εκτυπώνεται η εικόνα στην οθόνη με βάση τη συνάρτηση εκτύπωσης του βήματος 3.
- Εισάγοντας **L** ή **l** το πρόγραμμα εμφανίζει το μήνυμα "**Enter file path to load image:**" και στη συνέχεια περιμένει από το χρήστη να εισάγει ένα έγκυρο file path από το οποίο θα διαβάσει μία νέα εικόνα. Στη συνέχεια, χρησιμοποιεί τη συνάρτηση **load_image** για να διαβάσει την τρέχουσα εικόνα στο file path που έδωσε ο χρήστης. Εάν η συνάρτηση **load_image** αποτύχει, εκτυπώνει το μήνυμα "**Invalid path!**" ακολουθούμενο από χαρακτήρα αλλαγής γραμμής και επαναλαμβάνει τη διαδικασία μέχρι ο χρήστης να δώσει ένα έγκυρο path. Μετά το επιτυχημένο διάβασμα της εικόνας εκτυπώνει το μήνυμα "**Load OK!**" ακολουθούμενο από χαρακτήρα αλλαγής γραμμής.
- Εισάγοντας **S** ή **s** το πρόγραμμα εμφανίζει το μήνυμα "**Enter file path to save image:**" και στη συνέχεια περιμένει από το χρήστη να εισάγει ένα έγκυρο file path στο οποίο θα γράψει την τρέχουσα εικόνα. Στη συνέχεια, χρησιμοποιεί τη συνάρτηση **save_image** για να γράψει την τρέχουσα εικόνα στο file path που έδωσε ο χρήστης. Εάν η συνάρτηση **save_image** αποτύχει, εκτυπώνει το μήνυμα "**Invalid path!**" ακολουθούμενο από χαρακτήρα αλλαγής γραμμής και επαναλαμβάνει τη διαδικασία μέχρι ο χρήστης να δώσει ένα έγκυρο path. Μετά την επιτυχημένη αποθήκευση εκτυπώνει το μήνυμα "**Save OK!**" ακολουθούμενο από χαρακτήρα αλλαγής γραμμής.
- Εισάγοντας **Q** ή **q** το πρόγραμμα τερματίζει.

Βήμα 5ο – Μεταβολή της φωτεινότητας της εικόνας

Μπορείτε να μεταβάλλετε τη φωτεινότητα της εικόνας σε πιο ανοιχτή ή πιο σκούρα εάν πολλαπλασιάσετε τη φωτεινότητα και των τριών χρωμάτων όλων των εικονοστοιχείων της εικόνας με τον ίδιο αριθμό. Για τις ανάγκες του προγράμματος μας, επιτρέπεται στον χρήστη ο πολλαπλασιασμός των τιμών όλων των εικονοστοιχείων με ένα αριθμό θετικό και μικρότερο ή ίσο του 2.

Γράψτε μία συνάρτηση η οποία λαμβάνει ως παραμέτρους ένα πίνακα ακεραίων δύο διαστάσεων, έναν ακέραιο που συμβολίζει το πλάτος της εικόνας, έναν ακέραιο που συμβολίζει το ύψος της εικόνας και έναν ακέραιο που συμβολίζει τη μέγιστη τιμή της φωτεινότητας για όλα τα χρώματα RGB και έναν αριθμό κινητής υποδιαστολής που περιγράφει το συντελεστή μεταβολής της φωτεινότητας με βάση τον οποίο η συνάρτηση θα πολλαπλασιάσει τις τιμές φωτεινότητας όλων των εικονοστοιχείων και για τα τρία RGB χρώματα. Η συνάρτηση επιστρέφει **void**.

Σημείωση: Εάν η συνάρτηση υπολογίζει τιμές φωτεινότητας μεγαλύτερες της μέγιστης, αυτές θα πρέπει να αποκόπτονται στην τιμή της μέγιστης φωτεινότητας.

Στη συνάρτηση **main**, συμπληρώστε κώδικα ώστε το κεντρικό μενού του προγράμματος να είναι

Enter option:

Print image to screen: (P/p)
Load another image: (L/l)
Save image: (S/s)
Change luminosity: (U/u)
Quit: (Q/q)

Στη συνάρτηση **main** θα πρέπει να προσθέσετε μία επιπλέον επιλογή προς τον τελικό χρήστη. Εισάγοντας **U** ή **u** το πρόγραμμα εμφανίζει το μήνυμα **"Enter luminosity factor (factor>0 && factor<=2): "** και στη συνέχεια διαβάζει έναν αριθμό κινητής υποδιαστολής που αντιπροσωπεύει το συντελεστή μεταβολής της φωτεινότητας. Εάν δοθεί τιμή εκτός του διαστήματος (0,2] το πρόγραμμα εκτυπώνει το μήνυμα **"Factor should be between (0,2]"**, ακολουθούμενο από χαρακτήρα αλλαγής γραμμής και επαναλαμβάνει την ερώτηση προς τον χρήστη μέχρι να δοθεί έγκυρη τιμή.

Εφόσον δοθεί έγκυρη τιμή, το πρόγραμμα σας χρησιμοποιεί τη συνάρτηση που γράψατε σε αυτό το βήμα προκειμένου να μεταβάλει τη φωτεινότητα της εικόνας με βάση τον συντελεστή που εισήγαγε ο χρήστης. Μετά την μετατροπή εκτυπώνει το μήνυμα **"Conversion OK!"** ακολουθούμενο από χαρακτήρα αλλαγής γραμμής.

Βήμα 6ο – Μετατροπή της εικόνας σε ασπρόμαυρη

Μια ασπρόμαυρη (*grayscale*) εικόνα είναι μία εικόνα της οποίας και τα τρία RGB χρώματα κάθε εικονοστοιχείου έχουν την ίδια τιμή. Για παράδειγμα οι RGB τιμές 0 0 0 αντιπροσωπεύουν το μαύρο χρώμα, οι τιμές 255 255 255 το λευκό και οι τιμές 128 128 128 τη **συγκεκριμένη απόχρωση του γκρι**.

Γράψτε μία συνάρτηση η οποία λαμβάνει ως παραμέτρους ένα πίνακα ακεραίων δύο διαστάσεων, έναν ακέραιο που συμβολίζει το πλάτος της εικόνας και έναν ακέραιο που συμβολίζει το ύψος της εικόνας. Για κάθε εικονοστοιχείο της εικόνας η συνάρτηση υπολογίζει την απόχρωση του γκρι με βάση την παρακάτω φόρμουλα. Η συνάρτηση έχει επιστρεφόμενο τύπο **void**.

$$\text{Gray} = (\text{Red} * 0.3 + \text{Green} * 0.59 + \text{Blue} * 0.11)$$

Στη συνάρτηση **main**, συμπληρώστε κώδικα ώστε το κεντρικό μενού του προγράμματος να είναι

Enter option:

Print image to screen: (P/p)
Load another image: (L/l)
Save image: (S/s)
Change luminosity: (U/u)
Convert to grayscale: (G/g)
Quit: (Q/q)

Στη συνάρτηση **main** θα πρέπει να προσθέσετε μία επιπλέον επιλογή προς τον τελικό χρήστη. Εισάγοντας **G** ή **g** το πρόγραμμα καλεί τη συνάρτηση που γράψατε παραπάνω και την μετατρέπει σε ασπρόμαυρη. Μετά την μετατροπή εκτυπώνει το μήνυμα **"Conversion OK!"** ακολουθούμενο από χαρακτήρα αλλαγής γραμμής.

Βήμα 7ο – Διπλασιασμός του μεγέθους της εικόνας

Γράψτε μία συνάρτηση η οποία λαμβάνει ως παραμέτρους ένα πίνακα ακεραίων δύο διαστάσεων, ένα δείκτη σε ακέραιο που συμβολίζει το πλάτος της εικόνας και ένα δείκτη σε ακέραιο που συμβολίζει το ύψος της εικόνας και επιστρέφει **void**. Η συνάρτηση εξετάζει εάν η χωρητικότητα του

υφιστάμενου πίνακα (**IMG_SIZExIMG_SIZE**) επαρκεί για τον διπλασιασμό της εικόνας. Εφόσον επαρκεί διπλασιάζει τον πίνακα με βάση τον εξής αλγόριθμο.

Η φωτεινότητα του εικονοστοιχείου στη θέση **row**, **col** αντιγράφεται στις θέσεις

- **2*row, 2*col,**
- **2*row+1, 2*col,**
- **2*row, 2*col+1**
- **2*row+1, 2*col+1**

Σημείωση: Ξεκινήστε να κάνετε τις μετατροπές από το τέλος προς την αρχή κάθε γραμμής και κάθε στήλης. ΔΕΝ επιτρέπεται να χρησιμοποιήσετε βοηθητικό πίνακα.

Στη συνάρτηση **main**, συμπληρώστε κώδικα ώστε το κεντρικό μενού του προγράμματος να είναι

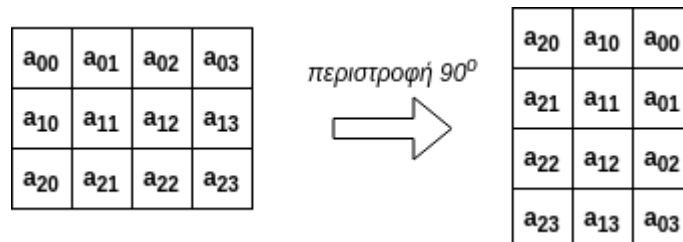
Enter option:

```
Print image to screen: (P/p)
Load another image: (L/l)
Save image: (S/s)
Change luminosity: (U/u)
Convert to grayscale: (G/g)
Duplicate size: (D/d)
Quit: (Q/q)
```

Στη συνάρτηση **main** θα πρέπει να προσθέσετε μία επιπλέον επιλογή προς τον τελικό χρήστη. Εισάγοντας **D** ή **d** το πρόγραμμα καλεί τη συνάρτηση που γράψατε παραπάνω και διπλασιάζει το μέγεθος της εικόνας. Μετά την μετατροπή εκτυπώνει το μήνυμα **"Duplication OK!"** ακολουθούμενο από χαρακτήρα αλλαγής γραμμής.

Βήμα 8ο – Δεξιά περιστροφή της εικόνας κατά 90°

Γράψτε μία συνάρτηση η οποία λαμβάνει ως παραμέτρους ένα πίνακα ακεραίων δύο διαστάσεων, ένα δείκτη σε ακέραιο που συμβολίζει το πλάτος της εικόνας και ένα δείκτη σε ακέραιο που συμβολίζει το ύψος της εικόνας και επιστρέφει **void**. Η συνάρτηση περιστρέφει δεξιόστροφα την εικόνα. Μπορείτε να εξάγετε τον αλγόριθμο περιστροφής από την μεταβολή στη θέση των στοιχείων της παρακάτω εικόνας. Για την περιστροφή μπορείτε να χρησιμοποιήσετε βοηθητικό πίνακα μέσα στη συνάρτηση που θα γράψετε.



Στη συνάρτηση **main**, συμπληρώστε κώδικα ώστε το κεντρικό μενού του προγράμματος να είναι

Enter option:

```
Print image to screen: (P/p)
Load another image: (L/l)
Save image: (S/s)
Change luminosity: (U/u)
Convert to grayscale: (G/g)
Duplicate size: (D/d)
Rotation: (R/r)
Quit: (Q/q)
```


Στη συνάρτηση **main** θα πρέπει να προσθέσετε μία επιπλέον επιλογή προς τον τελικό χρήστη. Εισάγοντας **R** ή **r** το πρόγραμμα καλεί τη συνάρτηση που γράψατε παραπάνω και περιστρέφει την εικόνα κατά 90° . Μετά την μετατροπή εκτυπώνει το μήνυμα "**Rotation OK!**" ακολουθούμενο από χαρακτήρα αλλαγής γραμμής.

Πώς να παραδώσετε τη δουλειά σας

Πριν παραδώσετε τα προγράμματά σας, προσθέστε σε σχόλια στην αρχή του κάθε αρχείου C τα πλήρη ονόματα και AEM των μελών της ομάδας. Παρακαλούμε να γράφετε τα σχόλια ΜΟΝΟ με λατινικούς χαρακτήρες.

Κατασκευάστε ένα φάκελο με όνομα `hw3_epwnumero1_AEM1_epwnumero3_AEM2` και αντιγράψτε μέσα σε αυτόν το `hw3.c`. Μη συμπεριλάβετε εικόνες στο φάκελο, αλλά μόνο το `hw3.c`.

Πηγαίνετε στο φάκελο μέσα στον οποίο βρίσκεται το `hw3_epwnumero1_AEM1_epwnumero2_AEM2` που κατασκευάσατε και γράψτε την παρακάτω εντολή:

```
tar czvf hw3_epwnumero1_AEM1_epwnumero2_AEM2.tgz hw3_epwnumero1_AEM1_epwnumero2_AEM2
```

Στείλτε email:

- στη διεύθυνση CE120LAB@gmail.com
- αντίγραφο (CC) στον άλλο μέλος της ομάδας σας
- θέμα (subject) **CE120 hw3**
- και συνημμένο αρχείο το `hw3_epwnumero1_AEM1_epwnumero2_AEM2.tgz`

Επιβεβαιώστε ότι το άλλο μέλος της ομάδας έλαβε αντίγραφο και είναι σωστό.