

I. ΑΝΤΙΓΡΑΦΗ ΔΥΑΔΙΚΟΥ ΑΡΧΕΙΟΥ

Σε αυτή την εργασία καλείστε να επεξεργαστείτε δυαδικά αρχεία με τις συναρτήσεις `open()`, `read()`, `write()`, `close()`. Συγκεκριμένα πρέπει να φτιάξετε μία συνάρτηση `copy()` η οποία δέχεται ως ορίσματα τα ονόματα δύο (δυαδικών) αρχείων και αντιγράφει το πρώτο στο δεύτερο.

Συγκεκριμένα, θα ανοίγει το πρώτο δυαδικό αρχείο για διάβασμα και το δεύτερο για δημιουργία (με μηδενισμό) και γράψιμο. Θα διαβάζει 1 byte τη φορά από το πρώτο και το γράφει στο δεύτερο αρχείο.

Η `main()` θα πρέπει να δέχεται τα ονόματα των δυο αρχείων ως ορίσματα.

Δοκιμάστε να αντιγράψετε το `a.out` σε ένα άλλο αρχείο (π.χ. `new.out`), το οποίο προφανώς πρέπει να είναι εκτελέσιμο. Αν όλα πάνε καλά θα πρέπει να μπορείτε να τρέξετε το νέο αρχείο όπως έτρεχε το `a.out`.

II. ΑΡΧΕΙΑ ΕΙΚΟΝΑΣ PNG

Στην άσκηση αυτή θα δημιουργήσετε ένα πρόγραμμα το οποίο βρίσκει και εμφανίζει βασικά στοιχεία για μία εικόνα τύπου PNG. Όλα τα αρχεία εικόνας αυτού του τύπου στα αρχικά bytes τους περιέχουν ορισμένες πληροφορίες ('μετα-πληροφορίες') σχετικά με την εικόνα που αποθηκεύουν.

Τα αρχεία PNG συγκεκριμένα, στα πρώτα τους 8 bytes πρέπει να αποθηκεύουν τους εξής 8 αριθμούς: 137, 80, 78, 71, 13, 10, 26, 10. Αυτοί αποτελούν την λεγόμενη «υπογραφή» PNG, και χρησιμοποιούνται στο να διαπιστώνεται αν ένα αρχείο είναι όντως PNG ή όχι.

Μετά από αυτά τα 8 bytes της υπογραφής, ακολουθούν πολλά τμήματα (chunks). Το πρώτο-πρώτο chunk είναι η κεφαλίδα (header), τα δεδομένα της οποίας καταλαμβάνουν ακριβώς 21 bytes ενώ τελευταίο είναι το τμήμα τέλους. Οι πληροφορίες που μας ενδιαφέρουν βρίσκονται στην κεφαλίδα.

Τα δεδομένα της κεφαλίδας έχουν με τη σειρά τις εξής πληροφορίες:

Όνομα	# bytes	Σχόλιο
info	8	(δεν μας ενδιαφέρει)
width	4	πλάτος εικόνας (σε pixel)
height	4	ύψος εικόνας (σε pixel)
bitdepth	1	(δεν μας ενδιαφέρει)
colortype	1	βλ. παρακάτω
compression	1	(δεν μας ενδιαφέρει)
filter	1	(δεν μας ενδιαφέρει)
interlace	1	(δεν μας ενδιαφέρει)

Παρατηρήσεις:

1. Όλοι οι ακέραιοι αριθμοί (4 bytes) αποθηκεύονται με **διάταξη μεγάλου άκρου (big endian)**. Θυμηθείτε πρώτον το προηγούμενο εργαστήριο, και δεύτερον, το γεγονός ότι τα συστήματα που δουλεύετε έχουν επεξεργαστές οι οποίοι χρησιμοποιούν αποθήκευση μικρού άκρου (little endian).
2. Το colortype χρησιμοποιείται ως εξής:

- αν το δεύτερο bit από δεξιά (πρώτο θεωρούμε το λιγότερο σημαντικό bit) είναι 1 τότε η εικόνα είναι έγχρωμη, αλλιώς είναι σε κλίμακα του γκρι.
- αν το τρίτο bit από δεξιά είναι 1 τότε η εικόνα έχει και διαφάνεια, αλλιώς όχι.

Ζητείται να φτιάξετε ένα πρόγραμμα το οποίο θα ανοίγει ένα αρχείο PNG, θα διαβάζει ορισμένες πληροφορίες από την κεφαλίδα του και θα τις εμφανίζει στην οθόνη. Συγκεκριμένα θα πρέπει να φτιάξετε μια `main` η οποία με χρήση ορισμάτων (`argc`, `argv`) θα δέχεται το όνομα ενός αρχείου PNG το οποίο και θα ανοίγει (ως δυαδικό) με την `open()`. Στη συνέχεια, κάνοντας χρήση των συναρτήσεων `read()` και `lseek()` θα διαβάζει τους κατάλληλους ακεραίους ώστε να υπολογίζει και να εμφανίσει τις εξής πληροφορίες:

1. Αν όντως είναι εικόνα PNG (ελέγχοντας την υπογραφή του αρχείου)
2. Πλάτος εικόνας σε pixel
3. Ύψος εικόνας σε pixel
4. Τύπος χρωματισμού (χρώμα ή κλίμακα γκρι)
5. Υπάρξη ή όχι διαφάνειας.

III. ΕΠΙΠΛΕΟΝ ΕΞΑΣΚΗΣΗ ΜΕ ΔΥΑΔΙΚΑ ΑΡΧΕΙΑ

Αναζητείστε πληροφορίες για δυαδικά αρχεία που σας φαίνονται ενδιαφέροντα, π.χ.

- αρχεία μουσικής (`mp3`, `avi`, κλπ)
- αρχεία βίντεο (`avi`, `3gp`, κλπ)
- αρχεία συστήματος (π.χ. η δομή ELF του αρχείου `a.out`)

Αφού βρείτε της πληροφορίες για το πώς είναι οργανωμένα τα αρχεία και οι κεφαλίδες τους, προσπαθήστε να φτιάξετε προγράμματα που τα ανοίγουν και εκτυπώνουν χρήσιμες πληροφορίες για αυτά.