

ΕΙΣΑΓΩΓΗ ΣΤΟΝ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟ (2022-23)

Εργασία 2

Το αντικείμενο της εργασίας αυτής είναι η επεξεργασία εικόνων που ακολουθούν το Netpbm format και η υλοποίηση συγκεκριμένων λειτουργιών επ' αυτών. Ο λόγος που υιοθετήθηκε το συγκεκριμένο format είναι ότι δεν απαιτείται η πλήρης φύλαξη των δεδομένων μίας εικόνας, ή έστω μέρους της, ώστε να είναι δυνατή η επεξεργασία της. Απλές λειτουργίες σε εικόνες που ακολουθούν αυτό το format μπορούν να υλοποιηθούν απλώς διαβάζοντας επαναλαμβανόμενα περιορισμένο πλήθος δεδομένων της εικόνας από την είσοδο, μέσω `getchar()`, χωρίς χρήση πινάκων για τη φύλαξη των δεδομένων αυτών.

Εικόνες και εικονοστοιχεία στον υπολογιστή

Ο τρόπος με τον οποίο παριστάνουμε εικόνες στον υπολογιστή είναι μέσω των λεγόμενων εικονοστοιχείων (pixels). Ουσιαστικά, μία εικόνα είναι μία ορθογώνια διάταξη εικονοστοιχείων, σε M γραμμές και N στήλες (οπότε το μέγεθος της εικόνας είναι $N \times M$, με το N να είναι το πλάτος της και το M το ύψος της), όπου για κάθε εικονοστοιχείο είναι καταχωρημένη η απαιτούμενη πληροφορία που προσδιορίζει πώς θα πρέπει να εμφανισθεί. Ένα από τα πολλά πρότυπα που υπάρχουν για την κωδικοποίηση εικόνων είναι το Netpbm format (<https://en.wikipedia.org/wiki/Netpbm>).

To Netpbm format

Το Netpbm format, υποστηρίζει, κατ' αρχήν, έγχρωμες εικόνες (δεδομένα σε μορφή ppm), όπου κάθε εικονοστοιχείο μπορεί να κωδικοποιηθεί από τρεις ακέραιους αριθμούς που αναπαριστούν το χρώμα του, σαν ποσότητες των βασικών χρωμάτων κόκκινο (R), πράσινο (G) και μπλε (B), εξ ου και τα αρχικά RGB που χρησιμοποιούμε όταν ασχολούμαστε με "χρωματικά" θέματα στον υπολογιστή. Συνήθως, αλλά όχι πάντοτε, οι αριθμοί αυτοί είναι μη προσημασμένοι ακέραιοι του ενός byte, δηλαδή κυμαίνονται από 0 έως 255. Το Netpbm format υποστηρίζει επίσης και ασπρόμαυρες εικόνες (δεδομένα σε μορφή pgm), όπου κάθε εικονοστοιχείο θα εμφανισθεί σε μία απόχρωση του γκρι, έχοντας κωδικοποιηθεί μέσω ενός μη προσημασμένου ακεραίου, συνήθως στο εύρος από 0 έως 255. Η τιμή 0 αντιστοιχεί στο απόλυτο μαύρο και η μέγιστη τιμή (συνήθως 255) στο απόλυτο άσπρο, ενώ οι ενδιάμεσες τιμές αντιστοιχούν σε κατάλληλες αποχρώσεις του γκρι. Τέλος, το Netpbm format υποστηρίζει και απολύτως ασπρόμαυρες εικόνες (δεδομένα σε μορφή pbm), όπου κάθε εικονοστοιχείο κωδικοποιείται με ένα bit, που είναι είτε το 0 (άσπρο) είτε το 1 (μαύρο).

Κωδικοποίηση εικονοστοιχείων

Μία δυνατότητα που δίνεται από το Netpbm format είναι η κωδικοποίηση των εικονοστοιχείων να είναι είτε δυαδική, είτε σε μορφή κειμένου (Ascii αναπαράσταση των ακεραίων τιμών). Όταν είναι δυαδική η κωδικοποίηση, για την αναπαράσταση ενός έγχρωμου εικονοστοιχείου χρειάζονται ακριβώς τρία bytes (ένα για κάθε χρώμα, υπό την προϋπόθεση ότι η μέγιστη τιμή για κάθε χρώμα δεν είναι μεγαλύτερη από 255). Αντίστοιχα, για τη δυαδική αναπαράσταση ενός εικονοστοιχείου σε απόχρωση του γκρι χρειάζεται ένα μόνο byte (πάλι με μέγιστη τιμή το 255), ενώ στην περίπτωση απολύτως ασπρόμαυρων εικόνων, μπορούμε να κωδικοποιήσουμε οκτώ εικονοστοιχεία με ένα byte, αφού για καθένα από αυτά χρειαζόμαστε ένα bit. Στην κωδικοποίηση των εικονοστοιχείων σε μορφή κειμένου, κάθε ακεραία τιμή παριστάνεται από τους Ascii κωδικούς των ψηφίων του αριθμού στο δεκαδικό σύστημα αρίθμησης. Για παράδειγμα, ενώ για τον αριθμό 230 χρειαζόμαστε ένα byte για τη δυαδική αναπαράστασή του, στην αναπαράσταση σε μορφή κειμένου χρειαζόμαστε τρία bytes, τους Ascii κωδικούς των ψηφίων 2, 3

και 0. Προφανώς, η δυαδική αναπαράσταση είναι πιο οικονομική σε μνήμη, αλλά η αναπαράσταση σε μορφή κειμένου είναι πιο εύκολα αναγνώσιμη από τον άνθρωπο.

Αναπαράσταση εικόνων

Η αναπαράσταση μίας εικόνας στο Netpbm format έχει την εξής μορφή:

- Αρχικά, υπάρχει ο “μαγικός αριθμός”, που χαρακτηρίζει τον τύπο της εικόνας και τη μορφή της κωδικοποίησης (δυαδική ή σε μορφή κειμένου). Συγκεκριμένα, αυτός μπορεί να είναι κάτι από τα ακόλουθα:

P1 (ασπρόμαυρη εικόνα σε μορφή κειμένου)
P2 (εικόνα με αποχρώσεις του γκρι σε μορφή κειμένου)
P3 (έγχρωμη εικόνα σε μορφή κειμένου)
P4 (ασπρόμαυρη εικόνα σε δυαδική κωδικοποίηση)
P5 (εικόνα με αποχρώσεις του γκρι σε δυαδική κωδικοποίηση)
P6 (έγχρωμη εικόνα σε δυαδική κωδικοποίηση)

Ο μαγικός αριθμός αναπαρίσταται από τους Ascii κωδικούς των δύο χαρακτήρων του.

- Ακολουθεί “λευκό διάστημα”, δηλαδή ένας ή περισσότεροι “λευκοί” χαρακτήρες (κενά, σπλη-γνώμονες, αλλαγές γραμμής).
- Μετά, υπάρχει το πλάτος της εικόνας, εκφρασμένο με τους Ascii κωδικούς των ψηφίων του στο δεκαδικό σύστημα και στη συνέχεια λευκό διάστημα.
- Στη συνέχεια, το ύψος της εικόνας, εκφρασμένο και αυτό με τους Ascii κωδικούς των ψηφίων του στο δεκαδικό σύστημα, όπως, δηλαδή, και το πλάτος και μετά, πάλι λευκό διάστημα.
- Για την περίπτωση έγχρωμων εικόνων ή εικόνων σε αποχρώσεις του γκρι, ακολουθεί η μέγιστη τιμή που μπορεί να έχει ένας αθέριος που κωδικοποιεί ένα χρώμα ή την απόχρωση του γκρι, αντίστοιχα, επίσης εκφρασμένος και αυτός με τους Ascii κωδικούς των ψηφίων του. Για την περίπτωση απολύτως ασπρόμαυρων εικόνων, δεν χρειάζεται κάποια σχετική πληροφορία, αφού οι τιμές που αντιστοιχούν σε ένα εικονοστοιχείο είναι καθορισμένες, 0 ή 1.
- Ακολουθεί ένας μοναδικός λευκός χαρακτήρας, στην περίπτωση δυαδικής κωδικοποίησης, ή λευκό διάστημα, στην περίπτωση κωδικοποίησης σε μορφή κειμένου.
- Τέλος, υπάρχουν τα δεδομένα της εικόνας κωδικοποιημένα κατάλληλα, ανάλογα με το είδος της εικόνας και τη μορφή της κωδικοποίησης. Η σειρά εμφάνισης των δεδομένων των εικονοστοιχειών είναι κατά γραμμές από επάνω προς τα κάτω και σε κάθε γραμμή, κατά στήλες από αριστερά προς τα δεξιά. Στις περιπτώσεις κωδικοποίησης σε μορφή κειμένου, οι αθέριοι που αναπαριστούν ποσότητα χρώματος ή απόχρωσης γκρι ή άσπρο/μαύρο εικονοστοιχείο διαχωρίζονται με λευκό διάστημα. Στην περίπτωση δυαδικής κωδικοποίησης ασπρόμαυρων εικόνων, όπου οκτώ εικονοστοιχεία κωδικοποιούνται σε ένα byte, αν το πλάτος της εικόνας δεν είναι πολλαπλάσιο του οκτώ, το τελευταίο byte **κάθε γραμμής** συμπληρώνεται με bits 1 (μαύρο εικονοστοιχείο).
- Μετά από το λευκό διάστημα που ακολουθεί τον μαγικό αριθμό και πριν από την πληροφορία για τις διαστάσεις της εικόνας, μπορεί να υπάρχουν και σχόλια. Ένα σχόλιο εκτείνεται από ένα χαρακτήρα '#' έως την επόμενη αλλαγή γραμμής ('\n'). Μπορεί να υπάρχουν σχόλια μετά από το λευκό διάστημα που ακολουθεί τον μαγικό αριθμό, στην ίδια γραμμή, αλλά μπορεί να υπάρχουν στη συνέχεια και γραμμές, μέχρι την πληροφορία πλάτους και ύψους της εικόνας, που να περιέχουν μόνο σχόλια.

Το ζητούμενο της εργασίας

Στην εργασία αυτή καλείσθε να υλοποιήσετε σε C ένα πρόγραμμα (πηγαίο αρχείο `figproc.c`) το οποίο θα μετατρέπει έγχρωμες εικόνες σε εικόνες με αποχρώσεις του γκρι και εικόνες με αποχρώσεις του γκρι σε ασπρόμαυρες εικόνες. Το πρόγραμμα θα πρέπει να διαβάζει τα δεδομένα της εικόνας που θα επεξεργασθεί με την `getchar()` από την πρότυπη είσοδο και να κάνει την κατάλληλη μετατροπή τους, το αποτέλεσμα της οποίας θα δίνει στην πρότυπη έξοδο. Το ποια μετατροπή θα γίνει, θα καθορίζεται από τον μαγικό αριθμό που διαβάζει αρχικά το πρόγραμμα από την είσοδο. Αν η είσοδος είναι έγχρωμη εικόνα, θα μετατρέπεται σε εικόνα με αποχρώσεις του γκρι. Αν η είσοδος είναι εικόνα με αποχρώσεις του γκρι, θα μετατρέπεται σε ασπρόμαυρη εικόνα. Το πρόγραμμα θα πρέπει να είναι σε θέση να μετατρέψει κατάλληλα τόσο δυαδικές κωδικοποιήσεις όσο και κωδικοποιήσεις σε μορφή κειμένου. Αν η είσοδος είναι σε δυαδική κωδικοποίηση, και η έξοδος θα είναι σε δυαδική κωδικοποίηση. Ομοίως, αν η είσοδος είναι σε μορφή κειμένου, και η έξοδος θα είναι σε μορφή κειμένου.

Για τη μετατροπή ενός έγχρωμου εικονοστοιχείου με τιμές στα τρία βασικά χρώματα R , G και B , να δοθεί στο αντίστοιχο εικονοστοιχείο απόχρωσης του γκρι, σύμφωνα με τη μέθοδο *φωτεινότητας* (luminosity), η τιμή:¹

$$\left\lfloor \frac{299 \cdot R + 587 \cdot G + 114 \cdot B}{1000} \right\rfloor$$

Για τη μετατροπή ενός εικονοστοιχείου απόχρωσης του γκρι σε απόλυτα άσπρο ή μαύρο εικονοστοιχείο, να ελέγχεται αν η τιμή της απόχρωσης του γκρι είναι μεγαλύτερη από το $\left\lfloor \frac{max+1}{2} \right\rfloor$, όπου max είναι η μέγιστη τιμή που μπορεί να έχει ένα εικονοστοιχείο της εισόδου. Αν είναι, τότε η έξοδος να είναι άσπρο εικονοστοιχείο (τιμή 0), αλλιώς μαύρο (τιμή 1).

Σε οποιαδήποτε περίπτωση μη αναμενόμενης εισόδου με βάση το `format` που περιγράφηκε, το πρόγραμμα να εκτυπώνει ένα διαγνωστικό μήνυμα (π.χ. `Input error!`) και να τερματίζει.

Κάποια παραδείγματα εκτέλεσης είναι τα εξής:

```
$ ./figproc < Nlbin.ppm > Nlbin.pgm
$ ./figproc < Nlbin.pgm > Nlbin.pbm
```

Για να δείτε τα περιεχόμενα αρχείων με δυαδική κωδικοποίηση, μπορείτε να χρησιμοποιήσετε την εντολή `od` (σε Unix-like συστήματα).² Με τις παρακάτω εντολές, βλέπουμε ανά 16άδες τα περιεχόμενα των αρχείων στο δεκαεξαδικό σύστημα, καθώς και τους χαρακτήρες που αντιστοιχούν στους κωδικούς αυτούς. Έχουν παραλειφθεί οι αναπαραστάσεις σε χαρακτήρες από το σημείο που αρχίζουν τα δεδομένα της εικόνας.

```
$ od -tx1c Nlbin.ppm
00000000  50  36  0a  23  20  43  72  65  61  74  65  64  20  62  79  20
          P   6  \n  #           C   r   e   a   t   e   d           b   y
00000020  49  72  66  61  6e  56  69  65  77  0a  32  30  34  38  20  31
          I   r   f   a   n   V   i   e   w   \n  2   0   4   8           1
00000040  35  33  36  0a  32  35  35  0a  4a  85  c9  49  86  c9  48  85
          5   3   6  \n  2   5   5  \n  .....
00000060  .....
$
```

¹ $\lfloor X \rfloor$ είναι το ακέραιο μέρος του X

²Για περιβάλλοντα Windows, υπάρχουν ελεύθερα διαθέσιμα προγράμματα που λειτουργούν παρόμοια με την εντολή `od` (π.χ. <https://www.hhdsoftware.com/free-hex-editor>).

```
$ od -tx1c NIbin.pgm
00000000 50 35 0a 32 30 34 38 20 31 35 33 36 0a 32 35 35
          P 5 \n 2 0 4 8      1 5 3 6 \n 2 5 5
00000020 0a 7b 7b 7a 77 74 6f 6b 6a 78 73 71 75 73 6f 6f
          \n .....
00000040 .....
$
$ od -tx1c NIbin.pbm
00000000 50 34 0a 32 30 34 38 20 31 35 33 36 0a ff ff ff
          P 4 \n 2 0 4 8      1 5 3 6 \n .....
00000020 .....
$
```

Οι εικόνες των προηγούμενων αρχείων είναι οι εξής:³



NIbin.ppm



NIbin.pgm



NIbin.pbm

Επίσης:

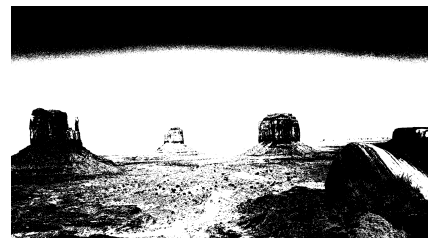
```
$ ./figproc < MVbin.ppm > MVbin.pgm
$ ./figproc < MVbin.pgm > MVbin.pbm
```



MVbin.ppm



MVbin.pgm



MVbin.pbm

³Τα αρχεία αυτά, καθώς και εκείνα που εμφανίζονται στη συνέχεια της εκφώνησης της εργασίας, μπορείτε να τα κατεβάσετε από το <http://www.di.uoa.gr/~ip/hwfiles/figproc/>, προσθέτοντας αμέσως μετά από αυτό το URL το όνομα του αρχείου που επιθυμείτε να κατεβάσετε. Για να εμφανίσετε αρχεία εικόνων σε Netpbm format σε περιβάλλον Windows, μπορείτε να κατεβάσετε το ελεύθερα διαθέσιμο πρόγραμμα Irfanview από το <http://www.irfanview.com/>.

Και ακόμα:

```
$ ./figproc < ISbin.ppm > ISbin.pgm  
$ ./figproc < ISbin.pgm > ISbin.pbm
```



ISbin.ppm



ISbin.pgm



ISbin.pbm

Τέλος, και μετατροπές αρχείων με κωδικοποίηση κειμένου:

```
$ ./figproc < colorsasc.ppm > colorsasc.pgm  
$ ./figproc < colorsasc.pgm > colorsasc.pbm
```



colorsasc.ppm



colorsasc.pgm



colorsasc.pbm

Επέκταση του προγράμματος με bonus βαθμολογία έως 30%

Σε περίπτωση που το ζητούμενο της εργασίας, όπως περιγράφηκε προηγουμένως, έχει υλοποιηθεί πλήρως και το πρόγραμμα λειτουργεί σωστά, προτείνεται να επεκτείνετε το πρόγραμμά σας ώστε να είναι σε θέση να κάνει και μετατροπές μεταξύ κωδικοποιήσεων (δυναμική και σε μορφή κειμένου) χωρίς αλλαγή του είδους της εικόνας (έγχρωμη, με αποχρώσεις του γκρι ή ασπρόμαυρη). Με άλλα λόγια, αυτό που ζητήθηκε ήδη στην εργασία είναι οι μετατροπές (με ορολογία μαγικών αριθμών) $P6 \rightarrow P5$, $P5 \rightarrow P4$, $P3 \rightarrow P2$ και $P2 \rightarrow P1$. Στην προτεινόμενη επέκταση μπορούν να υλοποιηθούν και οι μετατροπές $P6 \rightarrow P3$, $P3 \rightarrow P6$, $P5 \rightarrow P2$, $P2 \rightarrow P5$, $P4 \rightarrow P1$ και $P1 \rightarrow P4$. Ως προς τον τρόπο με τον οποίο θα καθορίζεται ποια μετατροπή θα κάνει το πρόγραμμά σας, έχετε την ελευθερία να ακολουθήσετε κάποια προσέγγιση της επιλογής σας. Η δέσμευση που υπάρχει είναι ότι το πρόγραμμα θα πρέπει να διαβάζει από την είσοδο και να εκτυπώνει στην έξοδο αυστηρά δεδομένα εικόνων και τίποτα άλλο. Οπότε, για παράδειγμα, αν το πρόγραμμα διαβάσει από την είσοδο εικόνα με μαγικό αριθμό $P6$, εσείς καθορίστε πώς θα επιλεγεί αν θα γίνει η μετατροπή $P6 \rightarrow P5$ (έγχρωμη εικόνα σε εικόνα με αποχρώσεις του γκρι, και οι δύο σε δυναμική κωδικοποίηση) ή η $P6 \rightarrow P3$ (έγχρωμη εικόνα δυναμικής κωδικοποίησης σε έγχρωμη εικόνα κωδικοποίησης κειμένου).

Σημειώσεις/Υποδείξεις/Απαγορεύσεις

- Προτείνεται να αναπτύξετε το πρόγραμμά σας σταδιακά. Δηλαδή, να υλοποιείτε τις ζητούμενες μετατροπές μία-μία και, αφού εξασφαλίζετε ότι τα αποτελέσματά σας είναι σωστά, να συνεχίζετε με την επόμενη.
- Στην εργασία αυτή απαγορεύονται αυστηρά η χρήση πινάκων (συμπεριλαμβανομένων και των συμβολοσειρών), συναρτήσεων της βιβλιοθήκης εισόδου-εξόδου της C που διαχειρίζονται αρχεία, αριθμών και μεταβλητών κινητής υποδιαστολής και συναρτήσεων της μαθηματικής βιβλιοθήκης της C.

- Η παράδοση της εργασίας αυτής συνίσταται στην υποβολή του πηγαίου αρχείου `figproc.c` μέσω του e-class του μαθήματος (επιλογή “Εργασίες”).
- Στην περίπτωση που αναπτύξετε το πρόγραμμά σας σε Windows μέσω Dev-C++, για την ορθή λειτουργία της ανάγνωσης από την είσοδο και της εκτύπωσης στην έξοδο δεδομένων σε δυαδική μορφή, πρέπει να συμπεριλάβετε στην αρχή του πηγαίου αρχείου σας το εξής τμήμα κώδικα:

```
#ifdef _WIN32
#include <io.h>
#include <fcntl.h>
#endif
```

και να συμπεριλάβετε στην αρχή της `main()` συνάρτησης το εξής τμήμα κώδικα:

```
#ifdef _WIN32
_setmode (_fileno (stdout), O_BINARY);
_setmode (_fileno (stdin), O_BINARY);
#endif
```

Στο τελικό σας παραδοτέο δεν υπάρχει λόγος να συμπεριληφθούν αυτά τα τμήματα κώδικα, δεδομένου ότι τα προγράμματά σας θα ελεγχθούν σε περιβάλλον Linux. Είναι αυτονόητο ότι θα ήταν καλό και εσείς να ελέγξετε την ορθή λειτουργία του προγράμματός σας σε περιβάλλον Linux πριν το υποβάλετε, για την αποφυγή πιθανών προβλημάτων.