

Άσκηση 3 ¹

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 $N \times M$), όπου για κάθε εικονοστοιχείο είναι καταχωρημένη η απαιτούμενη πληροφορία που προσδιορίζει πώς θα πρέπει να εμφανισθεί.

Σε έγχρωμες εικόνες, κάθε εικονοστοιχείο μπορεί να παρασταθεί από τρεις 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.

Μπορούμε, όμως, να έχουμε και ασπρόμαυρες εικόνες, για την ακρίβεια, εικόνες σε αποχρώσεις του γκρι, όπου κάθε εικονοστοιχείο παριστάνεται από έναν μόνο αριθμό (αρκεί να είναι του ενός byte, απροσημάστος όμως, δηλαδή από 0 έως 255), ο οποίος δείχνει το επίπεδο άσπρου-μαύρου (ή γκρι, αν θέλετε) του εικονοστοιχείου. Το 0 αντιστοιχεί στο απόλυτο μαύρο και το 255 στο απόλυτο άσπρο. Οι ενδιάμεσες τιμές αντιστοιχούν σε ανάλογες αποχρώσεις του γκρι. Μικρές τιμές είναι πιο μαύρα εικονοστοιχεία και μεγάλες πιο άσπρα.

Μία καθιερωμένη μορφή αρχείων που αναπαριστούν ασπρόμαυρες εικόνες είναι το λεγόμενο pgm (portable graymap) format. Με το εκτελέσιμο πρόγραμμα (για τα Suns του Τμήματος) που θα βρείτε στο <http://www.di.uoa.gr/~ip/hwprogs/pgmread>, μπορείτε εκτελώντας το με όρισμα ένα αρχείο σε pgm format να μετατρέψετε τα δεδομένα που είναι καταχωρημένα στο αρχείο σε μία ευκολότερα διαχειρίσιμη μορφή. Πάρτε το αρχείο <http://www.di.uoa.gr/~ip/images/demo.pgm> και δείτε τι αποτέλεσμα δίνει το πρόγραμμα pgmread όταν το διαβάσει:

```
% ./pgmread demo.pgm
8 16
255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255
255 255 255 255 255 150 150 150 150 150 150 255 255 255 255 255
255 255 150 150 150 150 255 255 255 255 150 150 150 150 255 255
255 255 182 182 150 255 0 0 0 0 255 150 182 182 255 255
255 255 182 182 150 255 0 0 0 0 255 150 182 182 255 255
255 255 150 150 150 150 255 255 255 255 150 150 150 150 255 255
255 255 255 255 255 150 150 150 150 150 150 255 255 255 255 255
255 255 255 255 255 255 255 255 255 255 255 255 255 255 255
%
```

Πρόκειται, δηλαδή για μία εικόνα μεγέθους 16×8 με τιμές απόχρωσης του γκρι στα εικονοστοιχεία αυτές που έδωσε στην έξοδο το πρόγραμμα pgmread. Το πρόγραμμα αυτό υπάρχει διαθέσιμο σε εκτελέσιμη μορφή και για Windows (pgmread-win.exe), αλλά και για Linux (pgmread-lin), στην ίδια θέση που βρίσκεται το pgmread. Επίσης υπάρχει διαθέσιμο και το αντίστροφο πρόγραμμα (pgmwrite, pgmwrite-win.exe και pgmwrite-lin) το οποίο, όταν διαβάσει από την είσοδο την περιγραφή μίας ασπρόμαυρης εικόνας στην απλοποιημένη μορφή που δίνει στην έξοδο το pgmread, γράφει στο αρχείο που του έχουμε δώσει σαν όρισμα την εικόνα σε pgm format. Για παράδειγμα, αν κάνουμε

¹Ευχαριστούμε πολύ τον Νίκο Καραμπατζιάκη, μεταπτυχιακό φοιτητή του Τμήματος, για την ιδέα, τα προγράμματα και τα αρχεία δεδομένων.

```
% ./pgmread demo.pgm > demo.txt
% ./pgmwrite newdemo.pgm < demo.txt
% ./pgmread newdemo.pgm > newdemo.txt
%
```

τότε τα αρχεία `demo.pgm` και `newdemo.pgm` έχουν ακριβώς τα ίδια περιεχόμενα, όπως και τα αρχεία `demo.txt` και `newdemo.txt`.

Για να δείτε τις εικόνες που αναπαριστούν αρχεία σε `pgm format`, αλλά και αρχεία σε άλλα `format`, και για να κάνετε μετατροπές από ένα `format` σε άλλο, μπορείτε να χρησιμοποιήσετε το πρόγραμμα `xv` στα `Suns` του Τμήματος (όχι όμως μέσω `putty`, αλλά δουλεύοντας απ' ευθείας σε κάποιο `Sun`), ή κάποιο πρόγραμμα από τα πολλά που υπάρχουν διαθέσιμα για `Windows`, π.χ. το `IrfanView`.

Αφού έχετε κατανοήσει πλήρως την παραπάνω, σχετικά εκτεταμένη, εισαγωγή, γράψτε ένα πρόγραμμα `C` (έστω ότι το πηγαίο αρχείο του ονομάζεται "`flood.c`") το οποίο να καλείται με τρία ακριβώς ορίσματα, π.χ. "`flood <row> <column> <value>`", και αφού διαβάσει από την είσοδο μία ασπρόμαυρη εικόνα στην απλοποιημένη μορφή που παράγει το πρόγραμμα `pgmread`, να αλλάζει το "χρώμα" στο εικονοστοιχείο που βρίσκεται στη γραμμή `<row>` και στήλη `<column>` (προσοχή, οι γραμμές και οι στήλες αριθμούνται από το 0) ώστε να έχει τιμή `<value>`, αλλά να "πλημμυρίζει" και όλα τα γειτονικά εικονοστοιχεία με το αρχικό, αν έχουν το ίδιο χρώμα με αυτό. Δύο εικονοστοιχεία είναι γειτονικά αν είναι συνεχόμενα, στην οριζόντια ή κατακόρυφη διεύθυνση. Η πλημμύρα αυτή να συνεχίζει και για όλα τα γειτονικά των γειτονικών με το ίδιο χρώμα του αρχικού, κ.ο.κ. Με αυτό τον τρόπο, όλη η περιοχή με το ίδιο χρώμα μέσα στην οποία βρίσκεται το εικονοστοιχείο (`<row>`, `<column>`) που δώσαμε στη γραμμή εντολής θα αλλάξει χρώμα σε `<value>`. Η έξοδος του προγράμματος να είναι στην ίδια μορφή με την είσοδο, δηλαδή στην απλοποιημένη εκδοχή του `pgm format` που διαχειρίζονται τα προγράμματα `pgmread` και `pgmwrite`. Κάποια παραδείγματα εκτέλεσης:

```
% ./pgmread demo.pgm > demo.txt
% ./flood 2 4 47 < demo.txt
8 16
255 255 255 255 255 255 255 255 255 255 255 255 255 255 255
255 255 255 255 255 47 47 47 47 47 47 255 255 255 255 255
255 255 47 47 47 47 255 255 255 255 47 47 47 47 255 255
255 255 182 182 47 255 0 0 0 0 255 47 182 182 255 255
255 255 182 182 47 255 0 0 0 0 255 47 182 182 255 255
255 255 47 47 47 47 255 255 255 255 47 47 47 47 255 255
255 255 255 255 255 47 47 47 47 47 47 255 255 255 255 255
255 255 255 255 255 255 255 255 255 255 255 255 255 255 255
% cat demo.txt | ./flood 6 13 0 | ./flood 4 7 255
8 16
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 150 150 150 150 150 150 0 0 0 0 0
0 0 150 150 150 150 255 255 255 255 150 150 150 150 0 0
0 0 182 182 150 255 255 255 255 255 150 182 182 0 0 0
0 0 182 182 150 255 255 255 255 255 150 182 182 0 0 0
0 0 150 150 150 150 255 255 255 255 150 150 150 150 0 0
0 0 0 0 0 150 150 150 150 150 150 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
%
```

Να δώσετε τη δυνατότητα στο πρόγραμμα που θα γράψετε να μπορεί να καλείται προαιρετικά και με μία τέταρτη παράμετρο `<epsilon>`, δηλαδή σαν "`flood <row> <column> <value> <epsilon>`",

όπου η λειτουργία του τότε να είναι η πλημμύρα με το χρώμα <value> όλων των εικονοστοιχείων που είναι μέσα σε μία περιοχή στην οποία βρίσκεται το εικονοστοιχείο (<row>,<column>) και των οποίων το χρώμα δεν διαφέρει περισσότερο από <epsilon> από το χρώμα του (<row>,<column>). Ουσιαστικά, η πρώτη εκδοχή του προγράμματος είναι ειδική περίπτωση της δεύτερης για <epsilon> ίσο με το μηδέν. Παραδείγματα:

```
% ./flood 2 13 10 40 < demo.txt
8 16
255 255 255 255 255 255 255 255 255 255 255 255 255 255 255
255 255 255 255 255 10 10 10 10 10 10 255 255 255 255 255
255 255 10 10 10 10 255 255 255 255 10 10 10 10 255 255
255 255 10 10 10 255 0 0 0 0 255 10 10 10 255 255
255 255 10 10 10 255 0 0 0 0 255 10 10 10 255 255
255 255 10 10 10 10 255 255 255 255 10 10 10 10 255 255
255 255 255 255 255 10 10 10 10 10 10 255 255 255 255 255
255 255 255 255 255 255 255 255 255 255 255 255 255 255 255
% ./pgmread demo.pgm | ./flood 0 0 0 | ./flood 3 4 255 110 | ./pgmwrite new.pgm
% ./pgmread new.pgm
8 16
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 255 255 255 255 255 255 0 0 0 0 0
0 0 255 255 255 255 255 255 255 255 255 255 255 0 0
0 0 255 255 255 255 0 0 0 0 255 255 255 255 0 0
0 0 255 255 255 255 0 0 0 0 255 255 255 255 0 0
0 0 255 255 255 255 255 255 255 255 255 255 255 0 0
0 0 0 0 0 255 255 255 255 255 255 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

Φροντίστε το πρόγραμμά σας να μην θέτει κάποιο περιορισμό στο μέγεθος των εικόνων που θα χειρίζεται, εκτός, βέβαια, από τη διαθέσιμη μνήμη του υπολογιστή σας.

Μπορείτε να πειραματιστείτε και με άλλα αρχεία σε pgm format που θα βρείτε κάτω από το <http://www.di.uoa.gr/~ip/images/>, όπως τα test.pgm, cartman.pgm, cross.pgm, αλλά και τα stsmall.pgm και stlarge.pgm.² Αν έχετε ή βρείτε και άλλες ενδιαφέρουσες εικόνες σε pgm format, στείλτε τις στο takis@di.uoa.gr ώστε να τοποθετηθούν στην παραπάνω θέση και να τις παίρνουν και οι συνάδελφοί σας για να πειραματίζονται.

Για την άσκηση αυτή, θα παραδώσετε ένα μόνο αρχείο πηγαίου κώδικα C, το flood.c, και τίποτε άλλο, σύμφωνα με τη διαδικασία υποβολής που περιγράφεται στη συνέχεια.