

# ΓΡΑΦΙΚΗ ΜΕ ΥΠΟΛΟΓΙΣΤΕΣ



## Πλήρωση Τριγώνων

Σιωππίδης Αθανάσιος 9090

8<sup>ο</sup> εξάμηνο

Έτος 2019-20 ΗΜΜΥ ΑΠΘ

## Λειτουργία και Τρόποι Κλίσης Προγραμμάτων

- `color = colorInterp(A,B,a,b,x)`

`a` = συντεταγμένη ενός άκρου

`b` = συντεταγμένη του αλλού άκρου

`A` = χρώμα στο σημείο `a`

`B` = χρώμα στο σημείο `b`

`x` = σημείο που θέλουμε να κάνουμε γραμμική παρεμβολή

**Λειτουργία:** Εκτελεί γραμμική παρεμβολή μεταξύ των χρωμάτων δύο σημείων.

**Κλίση:** Καλείται από την `triPaintGouraud`

- `demoFlat`

**Λειτουργία:** Εκτελεί τον χρωματισμό της πάπιας με flat τρόπο χωρίς ορίσματα και αποθηκεύει την εικόνα χρησιμοποιώντας την `imwrite` σε μορφή `png`

- `demoGouraud`

**Λειτουργία:** Εκτελεί τον χρωματισμό της πάπιας με Gouraud τρόπο χωρίς ορίσματα και αποθηκεύει την εικόνα χρησιμοποιώντας την `imwrite` σε μορφή `png`

- $I = \text{paintObject}(V, F, C, D, \text{painter})$

$V$  = Συντεταγμένες  $L$  κορυφών

$F$  = Δείκτης για τις κορυφές που αποτελούν κάθε τρίγωνο

$C$  = Χρώματα  $L$  κορυφών

$D$  = Βάθος  $L$  κορυφών

$\text{painter}$  = Τύπος χρωματισμού τριγώνου ("Flat", "Gouraud")

**Λειτουργία:** Χρωματίζει ένα σύνολο τριγώνων με τον τρόπο  $\text{painter}$  με σκοπό την δημιουργία κάποιας εικόνας

**Κλίση:** Καλείται από την **demoFlat & demoGouraud**

- $Y = \text{triPaintFlat}(X, V, C,)$

$X$  = Εικόνα στην οποία πρέπει να χρωματιστούν τα τρίγωνα

$V$  = Συντεταγμένες κορυφών τριγώνου

$C$  = Χρώματα κορυφών τριγώνου

**Λειτουργία:** Χρωματίζει ένα τρίγωνο με το τρόπο χρωματισμού Flat

- $Y = \text{triPaintGouraud}(X, V, C,)$

$X$  = Εικόνα στην οποία πρέπει να χρωματιστούν τα τρίγωνα

$V$  = Συντεταγμένες κορυφών τριγώνου

$C$  = Χρώματα κορυφών τριγώνου

**Λειτουργία:** Χρωματίζει ένα τρίγωνο με το τρόπο χρωματισμού Gouraud

# Περιγραφή Συναρτήσεων Χρωματισμού

## Τριγώνων

### tripaintFlat

Εστω  $k$ =πλευρά

1. Βρίσκουμε το μικρότερο και μεγαλύτερο  $y$  κάθε πλευράς
2. Σαν  $x_{kmin}$  παίρνουμε τα  $x$  που αντιστοιχούν στα  $y_{min}$  και  $y_{max}$  κάθε πλευράς
3. Βρίσκουμε το μεγαλύτερο και μικρότερο  $y$  (Δηλαδή από που έως που πάνε τα scanline)
4. Αρχικοποιούμε τις ενεργές πλευρές ως τις πλευρές που έχουν σαν  $y_{kmin}$  το μικρότερο  $y$  (σε περίπτωση οριζόντιας πλευράς ο πίνακας έχει 3 ενεργές πλευρές. Αυτό αντιμετωπίζεται στη συνέχεια)
5. Ως οριακά σημεία παίρνουμε τα  $x$  για τα οποία ισχύει ότι το  $y_{kmin}$  της πλευράς που ανήκουν, είναι ίσο με το μικρότερο  $y$
6. Μπαίνουμε στην επανάληψη για να βρούμε αναδρομικά τα οριακά σημεία για όλα τα scanline
  - a. Βλέπουμε αν βρισκόμαστε στην τελευταία σειρά pixel και αν αυτή είναι οριζόντια. Αν ισχύει, τότε γεμίζουμε τα pixel από το  $x_{kmin}:x_{kmax}$  μεταξύ των κορυφών της πλευράς που έχει  $y_{kmin}=y_{kmax}$  και βγαίνουμε από την επανάληψη
  - b. Αν απλά είμαστε στην τελευταία γραμμή τότε χρωματίζουμε το pixel της αντίστοιχης κορυφής και βγαίνουμε από την επανάληψη
  - c. Αν ο πίνακας ενεργών πλευρών έχει 3 γραμμές αντί για 2 τότε έχουμε οριζόντια γραμμή στην αρχή. Αφαιρούμε την οριζόντια πλευρά και ορίζουμε ως οριακά σημεία τα  $x$  που αντιστοιχούν στα μικρότερα  $y$  ( $y_{kmin}$ ) των 2 άλλων πλευρών
  - d. Χρωματίζουμε τα pixels μεταξύ των οριακών σημείων

- e. Αν το scanline είναι ίσο με το μέγιστο  $\gamma$  μιας πλευράς τότε αυτή η πλευρά αφαιρείται και στην θέση της μπαίνει η πλευρά που έχει ως μικρότερο  $\gamma$  το scanline αυτό
  - f. Βρίσκουμε την κλίση των ενεργών πλευρών
  - g. Βρίσκουμε τα οριακά σημεία από τα προηγούμενα οριακά σημεία προσθέτοντας  $1/a$  (όπου  $a$ =κλίση)
7. Δίνουμε την τιμή της εικόνας εισόδου με ζωγραφισμένο το τρίγωνο πάνω της στην έξοδο

### **triPaintGouraud**

Εστω  $k$ =πλευρά(υπάρχουν αλλαγές μόνο σε μερικά σημεία του προηγούμενου αλγόριθμου)

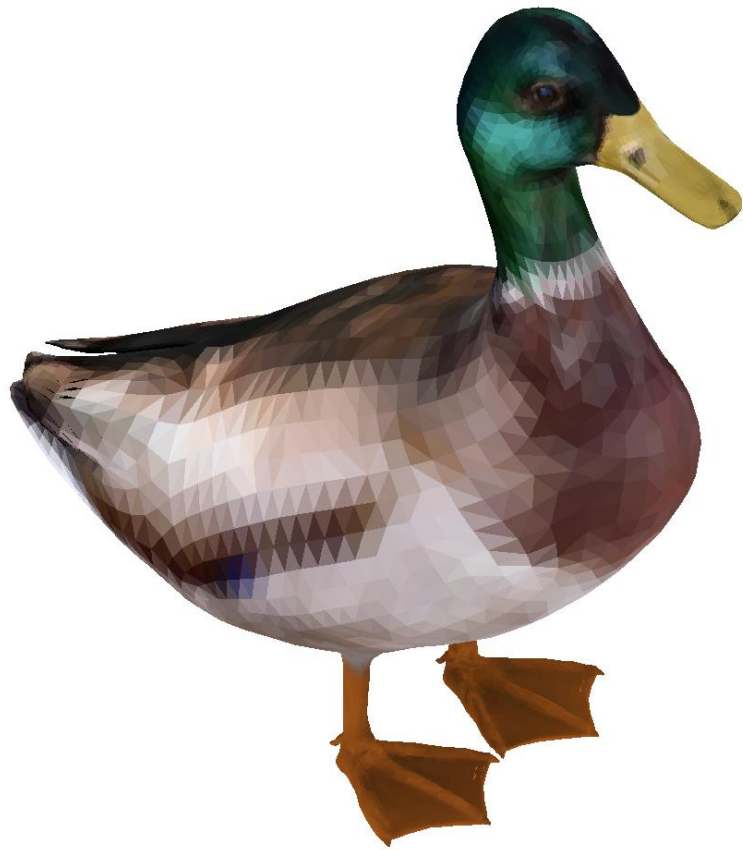
- 1. Βρίσκουμε το μικρότερο και μεγαλύτερο  $\gamma$  κάθε πλευράς
- 2. Σαν  $x_{kmin}$  παίρνουμε τα  $x$  που αντιστοιχούν στα  $\gamma_{min}$  και  $\gamma_{max}$  κάθε πλευράς καθώς και τα αντίστοιχα χρώματα
- 3. Βρίσκουμε το μεγαλύτερο και μικρότερο  $\gamma$  (Δηλαδή από που έως που πάνε τα scanline)
- 4. Αρχικοποιούμε τις ενεργές πλευρές ως τις πλευρές που έχουν σαν  $\gamma_{kmin}$  το μικρότερο  $\gamma$  (σε περίπτωση οριζόντιας πλευράς ο πίνακας έχει 3 ενεργές πλευρές. Αυτό αντιμετωπίζεται στη συνέχεια)
- 5. Ως οριακά σημεία παίρνουμε τα  $x$  για τα οποία ισχύει ότι το  $\gamma_{kmin}$  της πλευράς που ανήκουν, είναι ίσο με το μικρότερο  $\gamma$ , και αρχικοποιούμε έναν πίνακα με τα χρώματα τους.
- 6. Μπαίνουμε στην επανάληψη για να βρούμε αναδρομικά τα οριακά σημεία για όλα τα scanline
  - a. Βλέπουμε αν βρισκόμαστε στην τελευταία σειρά pixel και αν αυτή είναι οριζόντια. Αν ισχύει, τότε κάνουμε παρεμβολή στην κάθε ενεργή πλευρά για να βρούμε το χρώμα των οριακών σημείων. Στην συνέχεια γεμίζουμε τα pixel από το  $x_{kmin}:x_{kmax}$  μεταξύ των κορυφών της πλευράς που έχει  $\gamma_{kmin}=\gamma_{kmax}$  κάνοντας παρεμβολή για κάθε pixel μεταξύ των οριακών σημείων και βγαίνουμε από την επανάληψη

- b. Αν απλά είμαστε στην τελευταία γραμμή τότε χρωματίζουμε το pixel της αντίστοιχης κορυφής και βγαίνουμε από την επανάληψη
  - c. Αν ο πίνακας ενεργών πλευρών έχει 3 γραμμές αντί για 2 τότε έχουμε οριζόντια γραμμή στην αρχή. Αφαιρούμε την οριζόντια πλευρά και ορίζουμε ως οριακά σημεία τα  $x$  που αντιστοιχούν στα μικρότερα  $y(y_{kmin})$  των 2 άλλων πλευρών
  - d. Κάνουμε παρεμβολή στην κάθε ενεργή πλευρά για να βρούμε το χρώμα των οριακών σημείων.
  - e. Χρωματίζουμε τα pixels κάνοντας παρεμβολή μεταξύ των οριακών σημείων
  - f. Αν το scanline είναι ίσο με το μέγιστο  $y$  μιας πλευράς τότε αυτή η πλευρά αφαιρείται και στην θέση της μπαίνει η πλευρά που έχει ως μικρότερο  $y$  το scanline αυτό
  - g. Βρίσκουμε την κλίση των ενεργών πλευρών
  - h. Βρίσκουμε τα οριακά σημεία από τα προηγούμενα οριακά σημεία προσθέτοντας  $1/a$  (όπου  $a$ =κλίση)
7. Δίνουμε την τιμή της εικόνας εισόδου με ζωγραφισμένο το τρίγωνο πάνω της στην έξοδο

## **Παραδοχές**

- Όλα τα τρίγωνα είναι μέσα στο καμβά.
- Ο καμβάς είναι αντεστραμμένος δηλαδή τα σημεία αποτυπώνονται ως  $(y,x)$

**demoFlat(execution time=4.5 seconds)**



*demoGouraud(execution time=16.37*  
*seconds)*

