

## **ΓΡΑΦΙΚΗ ΜΕ ΥΠΟΛΟΓΙΣΤΕΣ 2022-2023**

Εργασία #1: Πλήρωση Τριγώνων

Καμπούρης – Μίχος Διόνυσος

7691

Συναρτήσεις που χρειάστηκαν για την υλοποίηση του προγράμματος:

- Συνάρτηση γραμμικής παρεμβολής  
`V = interpolate_vectors(p1, p2, V1, V2, xy, dim)`
- Συνάρτηση Πλήρωσης Τριγώνων με τη μέθοδο flat shading  
`updatedcanvas = flats(canvas, vertices, vcolors)`
- Συνάρτηση Πλήρωσης Τριγώνων με τη μέθοδο Gouraud shading  
`updatedcanvas = Gourauds(canvas, vertices, vcolors)`
- Συνάρτηση χρωματισμού αντικειμένου  
`img = render(verts2d, f aces, vcolors, depth, shade_t)`
- Συνάρτηση επιλογής μεθόδου πλήρωσης τριγώνων  
`img = shade_triangle(canvas, vertices, vcolors, shade_t)`

Η διαδικασία πλήρωσης τριγώνων ξεκινάει με το φόρτωμα των δεδομένων εισόδου και στην συνέχεια γίνεται κλήση της συνάρτησης *render*.

#### **ΚΛΗΣΗ ΣΥΝΑΡΤΗΣΗΣ `render(verts2d, f aces, vcolors, depth, shade_t)`**

Στην συνάρτηση *render* αρχικά ταξινομείτε ο πίνακας *faces*, στον οποίο είναι αποθηκευμένες οι συντεταγμένες των τριγώνων προς πλήρωση, σύμφωνα με το κέντρο βάρους των βαθών των κορυφών τους και στη συνέχεια καλείτε μέσα σε βρόγχο η συνάρτηση *shade\_triangle* για κάθε ένα από τα τρίγωνα ξεκινώντας από αυτά που βρίσκονται «βαθύτερα».

#### **ΚΛΗΣΗ ΣΥΝΑΡΤΗΣΗΣ `shade_triangle(canvas, vertices, vcolors, shade_t)`**

Η λειτουργία της *shade\_triangle* είναι πολύ απλή καθώς το μόνο που κάνει είναι να καλεί μια από τις δυο μεθόδους χρωματισμού των τριγώνων (flat ή Gouraud) ανάλογα με την τιμή της μεταβλητής *shade\_t*.

#### **Α΄ ΠΕΡΙΠΤΩΣΗ: ΜΕΘΟΔΟΣ ΧΡΩΜΑΤΙΣΜΟΥ flat (`shade_t="flat"`)**

Σε αυτή τη περίπτωση καλείται η συνάρτηση *flats* όπου το κάθε τρίγωνο βάφεται με ένα ενιαίο χρώμα το οποίο προκύπτει ως ο μέσος όρος των χρωμάτων των κορυφών του. Κατά τη διαδικασία χρωματισμού ενός τριγώνου ακολουθούνται τα εξής βήματα:

Αρχικά βρίσκουμε τις μέγιστες και ελάχιστες συντεταγμένες του τριγώνου ( $x_{min}$ ,  $x_{max}$ ,  $y_{min}$ ,  $y_{max}$ ), ελέγχουμε εάν τα μέγιστα με τα ελάχιστα συμπίπτουν και οδηγούμαστε σε τέσσερις περιπτώσεις:

1. Κάθετη γραμμή ( $x_{min}=x_{max}$ )
2. Οριζόντια γραμμή ( $y_{min}=y_{max}$ )
3. Σημείο ( $x_{min}=x_{max}$  και  $y_{min}=y_{max}$ )
4. Τρίγωνο (όταν δεν ισχύει τίποτα από τα παραπάνω)

Και τέλος χρωματίζουμε τα pixel ανάλογα με την περίπτωση.

### **Β' ΠΕΡΙΠΤΩΣΗ: ΜΕΘΟΔΟΣ ΧΡΩΜΑΤΙΣΜΟΥ Gouraud** (`shade_t="gouraud"`)

Στην δεύτερη περίπτωση καλείται η συνάρτηση *Gourauds* όπου το χρώμα υπολογίζεται ξεχωριστά για κάθε pixel και δίνεται μέσω της συνάρτησης *interpolate\_vectors* από την γραμμική παρεμβολή των χρωμάτων των σημείων τομής της ευθείας σάρωσης και των πλευρών του τριγώνου. Τα βήματα που ακολουθούνται πλην της επιλογής του χρώματος είναι τα ίδια με την συνάρτηση *flats* και φαίνονται στο κομμάτι ψευδοκώδικα που ακολουθεί.

```
for k=0:1:2
```

```
    βρίσκω τα  $x_{kmin}$ ,  $x_{kmax}$ ,  $y_{kmin}$ ,  $y_{kmax}$  της κάθε ακμής
```

```
end
```

```
 $y_{min} = \min_k (y_{0min}, y_{1min}, \dots)$ 
```

```
 $y_{max} = \max_k (y_{0max}, y_{1max}, \dots)$ 
```

```
 $x_{min} = \dots$ 
```

```
 $x_{max} = \dots$ 
```

Έλεγχος περίπτωσης (σημείο, ευθεία ή τρίγωνο) και χρωματισμός στην περίπτωση σημείου ή ευθείας

%4<sup>η</sup> περίπτωση: Τρίγωνο

Βρίσκουμε τη <Λίστα Ενεργών Ακμών> για τη Γραμμή Σάρωσης  $y = y_{min}$

Βρίσκουμε τη <Λίστα Ενεργών Οριακών Σημείων> για τη Γραμμή Σάρωσης  $y = y_{min}$

Διατάσσουμε τη <Λίστα Ενεργών Ακμών> με τις ακμές από αριστερά προς τα δεξιά  
Υπολογίζουμε την κλήση της κάθε πλευράς

*for*  $y = y_{min}:1:y_{max}$

Διατάσσουμε τη <Λίστα Ενεργών οριακών σημείων> ως προς  $x$

*for*  $x = x_{min}:1:x_{max}$

*drawpixel*( $x,y$ )

Ενημερώνουμε τη <Λίστα Ενεργών Ακμών>

Υπολογίζουμε τις νέες κλήσεις

Υπολογίζουμε τα νέα οριακά σημεία

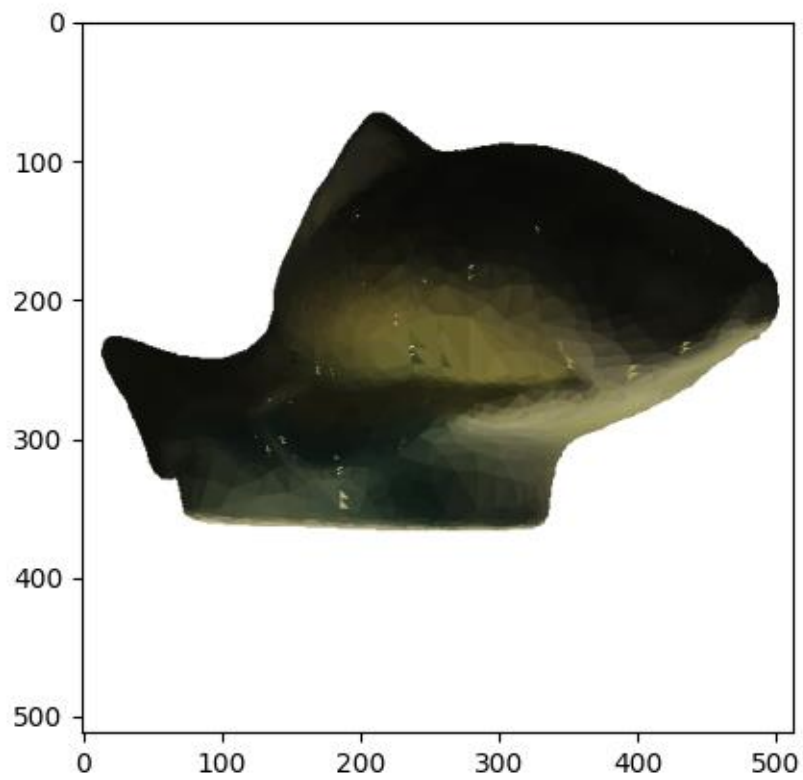
Ενημερώνουμε τη <Λίστα ενεργών οριακών σημείων>

*end*

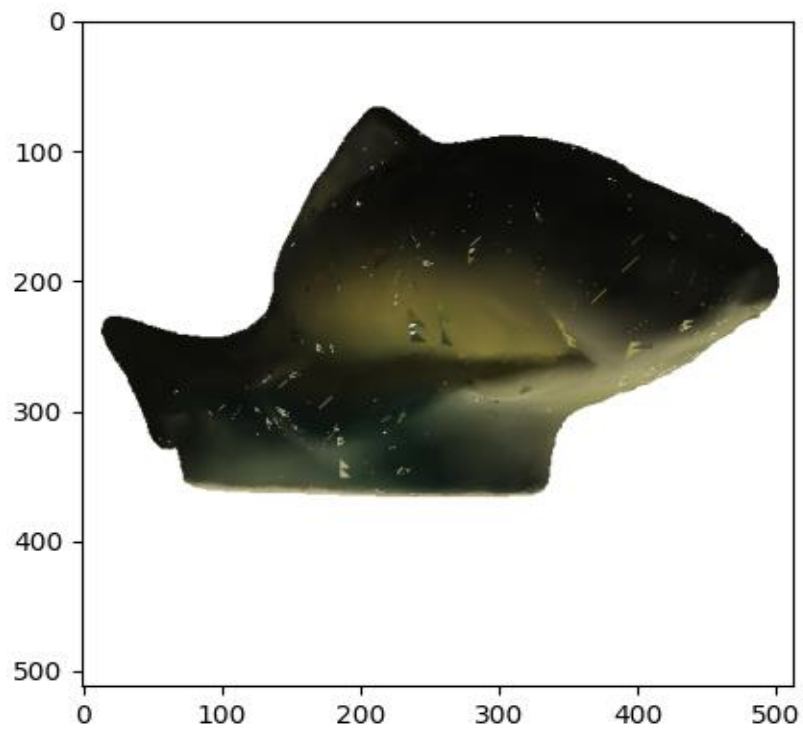
Η παραπάνω διαδικασία εκτελείται  $N$  φορές όπου  $N$  το πλήθος των τριγώνων.

Για τον χρωματισμό των *pixel* θεωρήσαμε ότι τα *pixel* που συμπίπτουν με οριακά σημεία ανήκουν στο δεξί ή και πάνω τρίγωνο.

## ΑΠΟΤΕΛΕΣΜΑΤΑ



*Εικόνα 1 μέθοδος flat shading*



*Εικόνα 2 μέθοδος Gouraud shading*

