



ΠΑΝΕΠΙΣΤΗΜΙΟ  
ΔΥΤΙΚΗΣ ΑΤΤΙΚΗΣ  
UNIVERSITY OF WEST ATTICA

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ  
ΥΠΟΛΟΓΙΣΤΩΝ

ΕΡΓΑΣΙΑ ΕΞΑΜΗΝΟΥ

ΣΧΗΜΑΤΙΣΜΟΣ ΤΡΙΣΔΙΑΣΤΑΤΗΣ ΣΚΗΝΗΣ  
ΓΡΑΦΙΚΩΝ ΜΕ ΧΡΗΣΗ ΤΗΣ WEBGL

**ΣΤΟΙΧΕΙΑ ΟΜΑΔΑΣ ΚΑΙ ΔΙΑΦΟΡΑ**

---

**ΣΤΟΙΧΕΙΑ ΦΟΙΤΗΤΗ 1:** ΑΘΑΝΑΣΙΟΥ ΒΑΣΙΛΕΙΟΣ ΕΥΑΓΓΕΛΟΣ (ΠΑΔΑ-19390005)

**ΣΤΟΙΧΕΙΑ ΦΟΙΤΗΤΗ 2:** ΤΑΤΣΗΣ ΠΑΝΤΕΛΗΣ (ΠΑΔΑ-20390226)

**ΥΠΕΥΘΥΝΟΣ ΜΑΘΗΜΑΤΟΣ:** ΜΠΑΡΔΗΣ ΓΕΩΡΓΙΟΣ

**ΗΜΕΡΟΜΗΝΙΑ ΥΠΟΒΟΛΗΣ:** 30/07/2024

**ΗΜΕΡΟΜΗΝΙΑ ΠΡΟΘΕΣΜΙΑΣ:** 30/07/2024

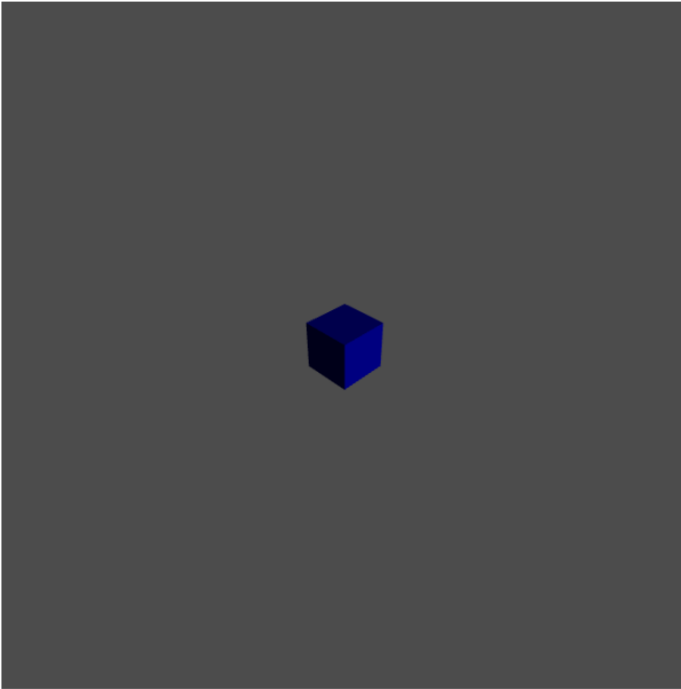
# ΓΡΑΦΙΚΑ ΥΠΟΛΟΓΙΣΤΩΝ

## ΠΕΡΙΕΧΟΜΕΝΑ

<b>Σκηνή 1 .....</b>	<b>3</b>
Βήμα 1 - Σχηματισμός Κύβου .....	3
Βήμα 2 - Τοποθέτηση Κάμερας .....	3
Βήμα 3 - Γωνία Θέασης / Ορθογώνια Απόσταση Κάμερας.....	4
Βήμα 4 - Κουμπί Επανασχεδίασης Σκηνής.....	4
<b>Σκηνή 2 .....</b>	<b>5</b>
Βήμα 5 - Μετασχηματισμός Κύβου σε Τραπεζί με χρώματα .....	5
Βήμα 6 - Μετασχηματισμός Κύβου σε Σκαμνί με χρώματα.....	5
Βήμα 7 - Μετασχηματισμός Σκαμνιού σε Καρέκλα με χρώματα .....	6
<b>Σκηνή 3 .....</b>	<b>7</b>
Βήμα 8 - Προσθήκη Animation .....	7
Βήμα 9 - Προσθήκη Textures αντί για χρώματα στα αντικείμενα .....	7
<b>Σκηνή 4 .....</b>	<b>9</b>
Βήμα 10 - Προσθήκη Skybox και 2D πατώματος.....	9
Βήμα 11 - Έλεγχος animation με το ποντίκι .....	9
Βήμα 12 - Ανατροπή καρέκλας με τη ροδέλα του ποντικιού.....	10
Βήμα 13 - Easter Egg .....	10

## Σκηνή 1

Ο πηγαίος κώδικας βρίσκεται στο αρχείο 1st\_scene.html



Γωνία θέασης (σε μοίρες):  Ορθογώνια απόσταση της κάμερας:

Θέση κάμερας: ☒ Left-Front-Top ☐ Left-Front-Bottom ☐ Left-Back-Top ☐ Left-Back-Bottom ☐ Right-Front-Top ☐ Right-Front-Bottom ☐ Right-Back-Top ☐ Right-Back-Bottom

### Βήμα 1 - Σχηματισμός Κύβου

Στο πρώτο βήμα με τη χρήση της εντολής `gl.clearcolor()`, εντός της `main` προσδώσαμε στο `background` το ζητούμενο χρώμα. Στην προκειμένη περίπτωση, δίνοντας στις μεταβλητές `R`, `G`, `B` ίδια τιμή έχουμε ως αποτέλεσμα μία απόχρωση του γκρι χρώματος. Για να σχεδιάσουμε τον κύβο ως ένα σύνολο τετραγώνων έπρεπε να θέσουμε έναν `buffer` για τις κορυφές, έναν για τα χρώματα και τέλος να δώσουμε σε αυτούς τις απαραίτητες τιμές. Η δυσκολία ήταν αναπόφευκτη εφόσον ήταν η αρχή της εργασίας, ωστόσο παίρνοντας ιδέες από την εργασία 6 του εργαστηρίου, που ήταν η πρώτη στην οποία ασχοληθήκαμε με 3D σχήματα καταφέραμε να ολοκληρώσουμε το πρώτο βήμα.

### Βήμα 2 - Τοποθέτηση Κάμερας

Στο δεύτερο βήμα της εργασίας, χρειάστηκε η χρήση των συναρτήσεων `lookAt()` και `perspective()` προκειμένου να μπορέσουμε να ορίσουμε τις απαραίτητες ρυθμίσεις στην κάμερα για να φαίνεται κατάλληλα ο κύβος που υλοποιήσαμε στο βήμα 1. Δεν υπήρξε κάποια δυσκολία στο συγκεκριμένο βήμα, καθώς οι τιμές των μεταβλητών που χρησιμοποιούν οι συναρτήσεις μας δίνονταν από την εκφώνηση.

# ΓΡΑΦΙΚΑ ΥΠΟΛΟΓΙΣΤΩΝ

## Βήμα 3 - Γωνία Θέασης / Ορθογώνια Απόσταση Κάμερας

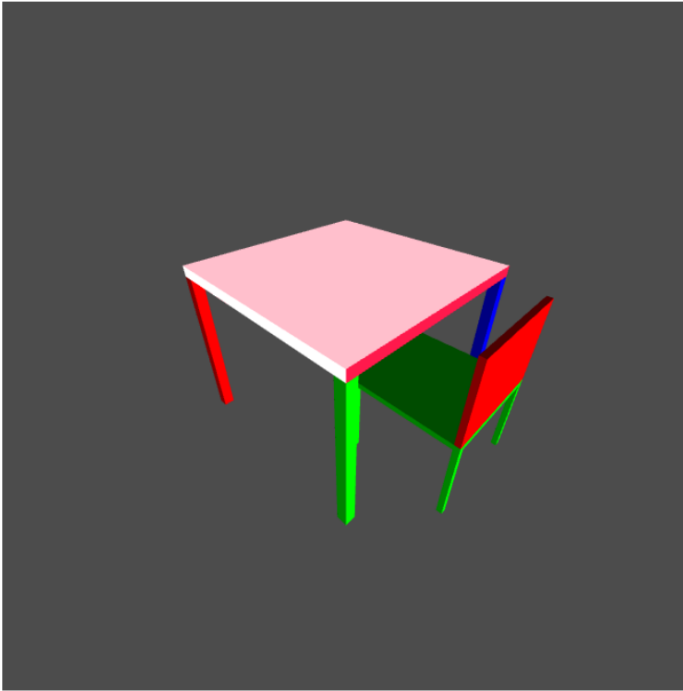
Στο τρίτο βήμα χρειάστηκε η τοποθέτηση text boxes για την γωνία θέασης και για την εισαγωγή της ορθογώνιας απόστασης. Επιτύχαμε την αλλαγή της θέσης της κάμερας ανάλογα με την επιλογή του χρήστη. Το πως αλλάζουν οι συντεταγμένες ήταν σαφές από την εκφώνηση, οπότε και σε αυτό το ερώτημα δεν αντιμετωπίστηκε κάποια σοβαρή δυσκολία.

## Βήμα 4 - Κουμπί Επανασχεδίασης Σκηνής

Για να υλοποιηθεί ωστόσο το βήμα 3 απαιτείται και κάποιο κουμπί που θα εκτελεί την συνάρτηση που επανασχεδιάζει τον κύβο μας ανάλογα με την επιθυμία του χρήστη. Τα δεδομένα από τα οποία θα ξαναδημιουργείται ο κύβος ανακτώνται από τα 2 text boxes που έχουμε φτιάξει. Το μόνο που χρειάζονταν να προσέξουμε αρκετά ήταν ότι έπρεπε να βάλουμε ένα flag ώστε οι προηγούμενες εντολές που αφορούν την τοποθέτηση της κάμερας αν μην τρέξουν.

## Σκηνή 2

Ο πηγαίος κώδικας βρίσκεται στο αρχείο 2nd\_scene.html



Γωνία θέασης (σε μοίρες):

Ορθογώνια απόσταση της κάμερας:

Θέση κάμερας: ☒ Left-Front-Top ☐ Left-Front-Bottom ☐ Left-Back-Top ☐ Left-Back-Bottom ☐ Right-Front-Top ☐ Right-Front-Bottom ☐ Right-Back-Top ☐ Right-Back-Bottom

### Βήμα 5 - Μετασχηματισμός Κύβου σε Τραπεζί με χρώματα

Για την υλοποίηση του βήματος 5 χρησιμοποιήσαμε αρκετά τον κώδικα από την σκηνή 1 ως βάση. Με την χρήση των συναρτήσεων `fromTranslation()`, `fromScaling()` και της `multiply()` επιτύχαμε αντίστοιχα την μετακίνηση, μεγέθυνση/σμίκρυνση και των πολλαπλασιασμό των πινάκων. Η συναρμολόγηση του τραπεζιού έγινε στην ουσία κομμάτι προς κομμάτι, οπότε κάθε κομμάτι είχε και διαφορετική επεξεργασία για την επίτευξη του. Η δυσκολία που αντιμετωπίστηκε στο συγκεκριμένο βήμα ήταν ότι χρειάστηκε να αλλάξουμε αρκετές τιμές προκειμένου όλα τα κομμάτια του τραπεζιού να συνδέονται μεταξύ τους, χωρίς κάποιο να “ξεκολλάει” από τα υπόλοιπα.

### Βήμα 6 - Μετασχηματισμός Κύβου σε Σκαμνί με χρώματα

Για την μετατροπή της τελικής καρέκλας, χρειάστηκε πρώτα η δημιουργία ενός σκαμνιού που είναι στην ουσία ένα τραπέζι με μικρότερες διαστάσεις. Για τον λόγο αυτό δεν χρειάστηκε η δημιουργία κάποιου νέου buffer για τις πλευρές ή τα χρώματα, αφού τους είχαμε ήδη διαθέσιμους για την κατασκευή του τραπεζιού. Δεν υπήρξε συνεπώς κάποια δυσκολία πέρα από το να βρούμε τις

## ΓΡΑΦΙΚΑ ΥΠΟΛΟΓΙΣΤΩΝ

σωστές τιμές μετακίνησης του σκαμνιού, στο συγκεκριμένο βήμα καθώς όλα τα εργαλεία ήταν διαθέσιμα.

### Βήμα 7 - Μετασχηματισμός Σκαμνιού σε Καρέκλα με χρώματα

Ουσιαστικά το μόνο που απαιτείται από το συγκεκριμένο βήμα είναι η προσθήκη ενός νέου κομματιού ως πλάτη της καρέκλας, του οποίου κομματιού τις διαστάσεις έχουμε από την εκφώνηση. Πάλι επειδή χρειάζονταν η τοποθέτηση ενός νέου κομματιού σε ένα ήδη υπάρχων αντικείμενο, χρειάστηκαν αρκετές δοκιμές ώστε το κομμάτι να “κουμπώνει” μαζί με την υπόλοιπη κατασκευή.

Στο πρόγραμμα ενδεχομένως να μην εμφανίζεται κατάλληλα το αρχικό frame και να χρειάζεται κάποιο interaction από τον χρήστη.

## Σκηνή 3

Ο πηγαίος κώδικας βρίσκεται στο αρχείο 3rd\_scene.html



Γωνία θέασης (σε μοίρες):  Ορθογώνια απόσταση της κάμερας:

Θέση κάμερας: ☒ Left-Front-Top ☐ Left-Front-Bottom ☐ Left-Back-Top ☐ Left-Back-Bottom ☐ Right-Front-Top ☐ Right-Front-Bottom ☐ Right-Back-Top ☐ Right-Back-Bottom

### Βήμα 8 - Προσθήκη Animation

Η τοποθέτηση των κουμπιών εκκίνησης και παύσης για το animation είναι κάτι το απλό που είχαμε δει και στα εργαστήρια. Ουσιαστικά χρειάστηκε πάλι πειραματισμός με τις τιμές της συνάρτησης `lookat()` για την περιστροφή της κάμερας και τις συναρτήσεις `Math.cos()` και `Math.sin()` για την υλοποίηση του αντικειμένου. Η υλοποίηση του συγκεκριμένου βήματος ήταν λίγο απαιτητική αλλά με τις κατάλληλες δοκιμές και σχεδιασμό ως προς τον τρόπο που θα καλούνται οι συναρτήσεις μεταξύ τους, επιτύχαμε το επιθυμητό αποτέλεσμα.

### Βήμα 9 - Προσθήκη Textures αντί για χρώματα στα αντικείμενα

Για το βήμα 9 συμβουλευτήκαμε την άσκηση 9 του εργαστηρίου που έχει να κάνει με textures και πως προσδίδουμε στα αντικείμενα μία πιο “ρεαλιστική” απεικόνιση. Μέχρι στιγμής ό,τι buffers είχαμε ήταν για χρώματα, συνεπώς χρειάστηκε να δημιουργήσουμε έναν νέο buffer για τα textures, ώστε να αντικαταστήσουμε τα buffers των χρωμάτων με αυτά. Οι διαστάσεις των εικόνων

## ΓΡΑΦΙΚΑ ΥΠΟΛΟΓΙΣΤΩΝ

επιλέχθηκαν στα 512x512 pixels (δύναμης του 2) και μορφής JPEG (.jpg), ώστε να μπορούμε να κάνουμε τεχνικές προεπεξεργασίας.

Το πρόβλημα που αντιμετωπίσαμε ήταν πάλι στις σωστές τιμές των texture buffers, καθώς, στην αρχή τα textures δεν φορτώναν στην σκηνή. Αντιθέτως, το τραπέζι με την καρέκλα είχανε μαύρο χρώμα.

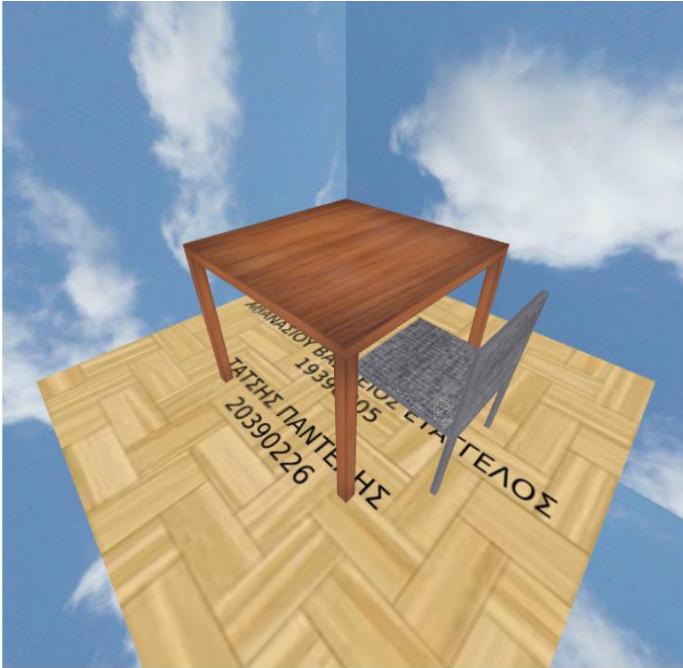
Άλλο πρόβλημα που αντιμετωπίσαμε ήταν στην προεπεξεργασία των εικόνων, αλλά με την κατάλληλη συμβουλή της άσκησης 9 του εργαστηρίου και την επιλογή διαστάσεων δύναμης του 2, καταφέραμε να το αντιμετωπίσουμε.

Για κάποιο λόγο στο πρώτο frame δεν υπάρχει κάποια εικόνα, προκειμένου να εμφανιστεί το επιθυμητό αποτέλεσμα χρειάζεται κάποιο interaction από τον χρήστη.



## Σκηνή 4

Ο πηγαίος κώδικας βρίσκεται στο αρχείο 4th\_scene.html



Γωνία θέασης (σε μοίρες):  Ορθογώνια απόσταση της κάμερας:

Θέση κάμερας: ☒ Left-Front-Top ☐ Left-Front-Bottom ☐ Left-Back-Top ☐ Left-Back-Bottom ☐ Right-Front-Top ☐ Right-Front-Bottom ☐ Right-Back-Top ☐ Right-Back-Bottom

### Βήμα 10 - Προσθήκη Skybox και 2D πατώματος

Για αυτό το βήμα 10 χρειάστηκε να ασχοληθούμε με τον ουρανό και το πάτωμα. Αρχικά τοποθετήσαμε τα αντικείμενα στη σκηνή ενεργοποιώντας πρώτα τα textures του ουρανού και σχεδιάζουμε το box με βάση τις οδηγίες που δίνει η εκφώνηση. Στη συνέχεια, ασχοληθήκαμε με το πάτωμα στο οποίο ήταν απαραίτητη η δημιουργία και χρήση νέων buffers όπως έναν indexBuffer και έναν textureBuffer λόγω ότι είναι 2D. Εφόσον το πάτωμα είναι 50x50 δεν ασχοληθήκαμε με την 3η διάσταση και μείναμε απλά στον σχεδιασμό ενός επίπεδου τετραγώνου.

Δεν υπήρξε κάποιο πρόβλημα, καθώς λόγω του texture buffer που δημιουργήσαμε τα ονόματα αναγράφονταν στο πάτωμα σωστά και καθαρά.

### Βήμα 11 - Έλεγχος animation με το ποντίκι

Ο έλεγχος με το ποντίκι στο βήμα 11 μας δημιούργησε αρκετά προβλήματα καθώς, παραλείπαμε αρκετά σημαντικά βήματα καθόλη τη διάρκεια δημιουργίας αυτής της λειτουργίας.

## ΓΡΑΦΙΚΑ ΥΠΟΛΟΓΙΣΤΩΝ

Ειδικά όσον αφορά τον έλεγχο με το ποντίκι όσο το animation είναι σταματημένο. Με τις κατάλληλες μετατροπές στον κώδικα ξεπεράστηκαν αυτά τα ζητήματα. Με την κλήση της συνάρτησης animationStep() μας βοηθάει στην ανανέωση της στιγμής. Αυτό που κάνουμε επί της ουσίας είναι κάθε φορά που κινείται το ποντίκι είναι να ανανεώσουμε τη σκηνή με τις όποιες τυχόν αλλαγές.

### Βήμα 12 - Ανατροπή καρέκλας με τη ροδέλα του ποντικιού

Στο βήμα αυτό αξιοποιήσαμε στοιχεία από την τελευταία άσκηση εργαστηρίου και ουσιαστικά με τη μέθοδο fromYRotation() πραγματοποιήσαμε περιστροφή της καρέκλας στον άξονα y. Επίσης, πραγματοποιήσαμε και αλλαγές στον πίνακα μετακίνησης με τη μέθοδο fromTranslation(), με αλλαγές στις τιμές του x και z άξονα της καρέκλας. Η γωνία περιστροφής μεταβάλλεται από την μεταβλητή wheelRadiusFactor, όπου καθορίζει το βήμα της ροδέλας του ποντικιού.

Προβλήματα που αντιμετωπίσαμε ήταν στις τιμές μετακίνησης της καρέκλας και του βήματος της ροδέλας του ποντικιού, καθώς, περιορίσαμε την καρέκλα να περιστρέφεται από 0 έως 90 μοίρες. Η πλάτη της καρέκλας είτε δεν ακουμπούσε το πάτωμα είτε το διαπερνούσε.

Άλλο πρόβλημα που αντιμετωπίσαμε ήταν ότι η καρέκλα μετακινούνταν και στους 3 άξονες, παρόλο που στο πίνακα translate αλλάζαμε 2 από τους 3 άξονες. Το πρόβλημα αντιμετωπίστηκε δίχως να αλλάζουμε την τιμή στον άξονα y στον πίνακα translate της καρέκλας, καθώς, ο wheelRadiusFactor επηρέαζε ήδη τον άξονα y με τη μέθοδο fromYRotation().

### Βήμα 13 - Easter Egg

Σε αυτό το βήμα μετράμε με δύο απλές μεταβλητές πόσες φορές έχει πέσει και έχει ανέβει η καρέκλα. Στη περίπτωση δηλαδή που γίνει αντιληπτό ότι η καρέκλα έχει πέσει τελείως (έχει τελέσει περιστροφή 90 μοιρών) το counter αυξάνεται κατά 1, όταν το counter αυτό είναι ίσο με το counter που μετράει το πόσες φορές έχει πέσει η καρέκλα. Έτσι γίνεται να αποφευχθεί οποιαδήποτε παραπάνω αύξηση του counter. Το counter που υπολογίζει τον αριθμό των φορών που σηκώνεται η καρέκλα και αυξάνεται μόνο στην περίπτωση που η αρχική θέση της καρέκλας είναι κάτω και το chairDown είναι μεγαλύτερο. Για την σχεδίαση επιπλέον καρέκλας για το Easter Egg, χρησιμοποιήσαμε τους ίδιους μετασχηματισμούς δημιουργίας της καρέκλας με αντεστραμμένα πρόσημα στις τιμές των x και y άξονα, ώστε να σχεδιαστεί ακριβώς απέναντι από την ήδη υπάρχουσα καρέκλα. Το σετ εντολών περιλαμβάνεται σε μια if που ελέγχει αν η καρέκλα έχει ανατραπεί 3 φορές.

Στο βήμα αυτό δεν αντιμετωπίστηκε κάποιο πρόβλημα, καθώς, τα βήματα υλοποίησης επαναλαμβάνονται από προηγούμενα βήματα.

# ΓΡΑΦΙΚΑ ΥΠΟΛΟΓΙΣΤΩΝ



Σας ευχαριστούμε για την προσοχή σας.

