

**ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΜΕ ΕΦΑΡΜΟΓΕΣ ΣΤΗΝ ΒΙΟΙΑΤΡΙΚΗ**

**2η ΕΡΓΑΣΙΑ ΕΞΑΜΗΝΟΥ**

**ΓΙΑ ΤΟ ΕΡΓΑΣΤΗΡΙΟ ΤΟΥ ΜΑΘΗΜΑΤΟΣ:**

**ΓΡΑΦΙΚΗ ΥΠΟΛΟΓΙΣΤΩΝ**

**ΘΕΜΑ:**

**ΜΕΤΑΣΧΗΜΑΤΙΣΜΟΣ ΘΕΑΣΗΣ ΜΕ ΤΗ ΧΡΗΣΗ ΤΗΣ ΔΙΕΠΑΦΗΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ ΕΦΑΡΜΟΓΩΝ ΤΗΣ OPENGL**

**ΕΚΠΟΝΗΘΗΚΕ ΑΠΟ ΤΟΥΣ ΦΟΙΤΗΤΕΣ:**

**ΚΩΝΣΤΑΝΤΙΝΙΔΗ ΠΑΥΛΟ, ΑΜ: 01651, 3o ΕΤΟΣ,**

**ΚΑΤΙΡΤΖΗ ΜΑΡΙΟ, ΑΜ: 01608, 3o ΕΤΟΣ**

**ΔΙΔΑΣΚΟΝΤΕΣ :**

**ΔΡΑΚΟΠΟΥΛΟΣ ΒΑΣΙΛΕΙΟΣ,**

**ΜΑΤΘΕΣ ΔΗΜΗΤΡΙΟΣ**

Λαμία, 2021

**ΠΕΡΙΕΧΟΜΕΝΑ**

* 1. **Περίληψη ....................................................................................................................... σελ. 3**
  2. **Abstract ….……………………………………………………………………………… p. 4**
  3. **Τεκμηρίωση Κώδικα .................................................................................................... σελ. 5**
  4. **Αναπαράσταση του Αποτελέσματός του Κώδικα .............................................. σελ. 11**
  5. **Συμπεράσματα ............................................................................................................. σελ. 12**
  6. **Επίλογος ....................................................................................................................... σελ. 12**
  7. **Βιβλιογραφία ............................................................................................................... σελ. 13**
  8. **Ευρετήριο .................................................................................................................... σελ. 14**
  9. **Περίληψη**

Η εργασία που επωμιστήκαμε αφορά τον μετασχηματισμό θέασης μιας τυχαίας σκηνής. Πιο συγκεκριμένα, χρησιμοποιήθηκε η προοπτική προβολή μιας σκηνής που περιέχει έναν πολύχρωμο κύβο. Η προοπτική προβολή εμφανίζει πολλά κοινά στοιχεία με την ανθρώπινη όραση. Το κέντρο της προβολής αντιστοιχεί στην ίριδα, ενώ το επίπεδο προβολής αντιστοιχεί στον αμφιβληστροειδή χιτώνα (Μουστάκας κ.α., 2015, σελ. 71-72). Γενικά, αυτού του είδους η προβολή δίνει ρεαλισμό στην προβαλλόμενη σκηνή, καθώς προσομοιώνεται η εικόνα μιας κάμερας. Για να επιτευχθεί αυτό, χρειάζεται να λάβουμε υπόψη ότι οι ακτίνες του φωτός που αντανακλώνται από τα αντικείμενα σε μια σκηνή, ακολουθούν τα ίχνη που συγκλίνουν προς το επίπεδο της ταινίας της κάμερας. Τα γεωμετρικά αυτά εφέ, μπορούν να προσεγγιστούν προβάλλοντας αντικείμενα στο επίπεδο θέασης κατά μήκος των ιχνών που συγκλίνουν σε μια θέση που ονομάζεται σημείο αναφοράς της προβολής (projection reference point) (ή κέντρο προβολής – center of projection). Κατόπιν, τα αντικείμενα προβάλλονται με εφέ προοπτικής, και οι προβολές των μακρινών αντικειμένων είναι μικρότερες από τις προβολές αντικειμένων του ίδιου μεγέθους που βρίσκονται πλησιέστερα στο επίπεδο θέασης(Hearn & Baker, 2004, σελ. 352). Τέλος, αν η κάμερα απομακρυνθεί σημαντικά από τον κύβο, τότε ο κύβος αποκόπτεται από την σκηνή και δεν προβάλλεται.

Ο κώδικας που υλοποιήθηκε, αποτελεί ένα διαδραστικό πρόγραμμα, δηλαδή, ο χρήστης μπορεί εύκολα με την χρήση των κουμπιών του πληκτρολογίου να χειριστεί την κάμερα και να παρατηρήσει από διαφορετικές οπτικές γωνίες τον κύβο, πειραματιζόμενος με τις συντεταγμένες της θέσης της κάμερας. Ειδικότερα, πατώντας τα κουμπιά ‘x’, ‘y’, ‘z’, ‘X’, ’Y’, ’Z’, μπορεί να αυξομειώσει την θέση της κάμερας ως προς τις συντεταγμένες x, y, z. Αναλόγως, μεταβάλλεται η οπτική γωνία που ο χρήστης βλέπει τον κύβο.

***Λέξεις Κλειδιά:*** *προοπτική προβολή, σκηνή, κάμερα, κύβος, επίπεδο θέασης*

*1 Ο αναγνώστης μπορεί να αναζητήσει την αναλυτικότερη επεξήγηση των συναρτήσεων στον ιστότοπο:* [*https://www.khronos.org/registry/OpenGL-Refpages/*](https://www.khronos.org/registry/OpenGL-Refpages/)

* 1. **Abstract**

The work we have done is concerning the viewing transformation of a random scene. More specifically, it concerns the perspective view of a scene containing a colorful cube. Perspective view, shares many things in common with human vision. The center of projection corresponds to the iris, while the level of projection corresponds to the retina (Moustakas et al., 2015, pp. 71-72). In general, this type of projection gives realism to the projected scene, because of the camera-oriented image is simulated. To achieve this, we need to keep in mind that the rays of light reflected by objects in a scene, follow the traces that converge to the level of the camera film. These geometric effects can be accessed by projecting objects at the viewing level along the tracks that converge at a position called the projection reference point (or center of projection). Objects are then projected with perspective effects, and the projections of distant objects are smaller than the projections of objects of the same size closest to the viewing level (Hearn & Baker, 2004, p. 352). Finally, if the camera moves significantly away from the cube, then the cube is cut off from the scene and not projected.

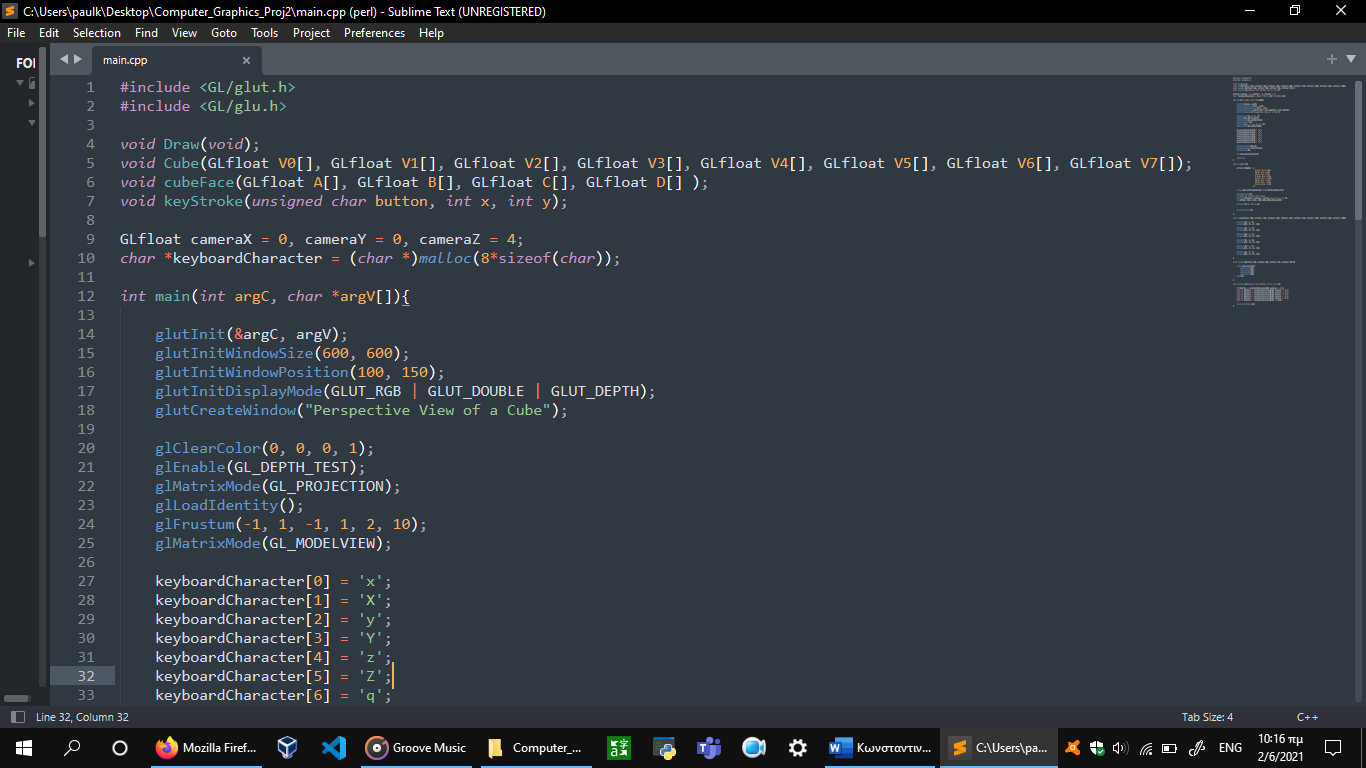
The implemented code is an interactive program, that is, the user can easily use the keyboard buttons to operate the camera and observe the cube from different angles, experimenting with the coordinates of the camera position. In particular, pressing the ‘x’, ‘y’, ‘z’, ‘X’, ‘Y’, ‘Z’ buttons can change the position of the camera relatively to the x, y, z coordinates. Accordingly, the angle of view that the user sees the cube changes.

***Keywords:*** *perspective view, scene, camera, cube, viewing level*

*1 The reader can search for the more detailed explanation of the functions of the OpenGL API on the website:* [*https://www.khronos.org/registry/OpenGL-Refpages/*](https://www.khronos.org/registry/OpenGL-Refpages/)

* 1. **Τεκμηρίωση Κώδικα**

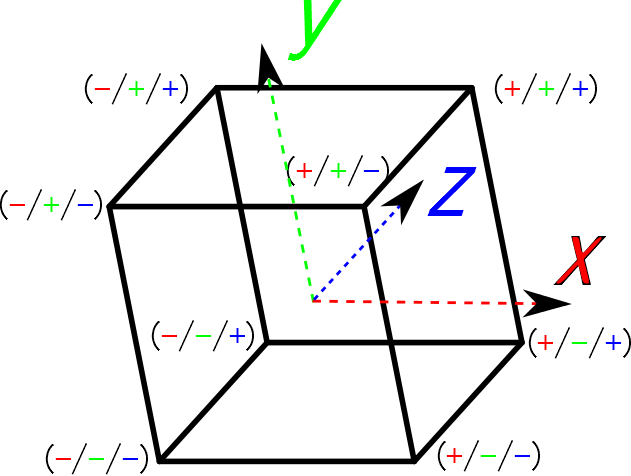
Αρχικά, με την εντολή του μικροεπεξεργαστή #include, εισάγουμε τις βιβλιοθήκες της διεπαφής προγραμματισμού εφαρμογών της OpenGL (glut.h και glu.h) (εικόνα 1, γραμμές 1, 2), οι οποίες θα μας επιτρέψουν να χρησιμοποιήσουμε χρήσιμες συναρτήσεις για την γραφική αναπαράσταση του κύβου αλλά και για την μετακίνηση της οπτικής γωνίας της κάμερας (προοπτική προβολή). Έπειτα, ακολουθούν οι 4 συναρτήσεις χρήστη που θα χρησιμοποιηθούν στο πρόγραμμα. Στις γραμμές 9, 10 ορίζονται κάποιες καθολικές μεταβλητές. Οι μεταβλητές cameraX, cameraY, cameraZ, χρησιμοποιούνται για την μετακίνηση της κάμερας ως προς τους άξονες Χ, Υ και Ζ αντίστοιχα.



Εικόνα 1

Ακολουθεί η περιγραφή, με τη σειρά, των τεσσάρων συναρτήσεων:

* Στη συνάρτηση **Draw()**, πρώτα από όλα, ορίζουμε έναν δισδιάστατο πίνακα 8x3 τύπου GLfloat, που αναπαριστά τις συντεταγμένες των 8 κορυφών του κύβου όπως αυτός αναπαρίσταται στον τρισδιάστατο χώρο (εικόνα 2).

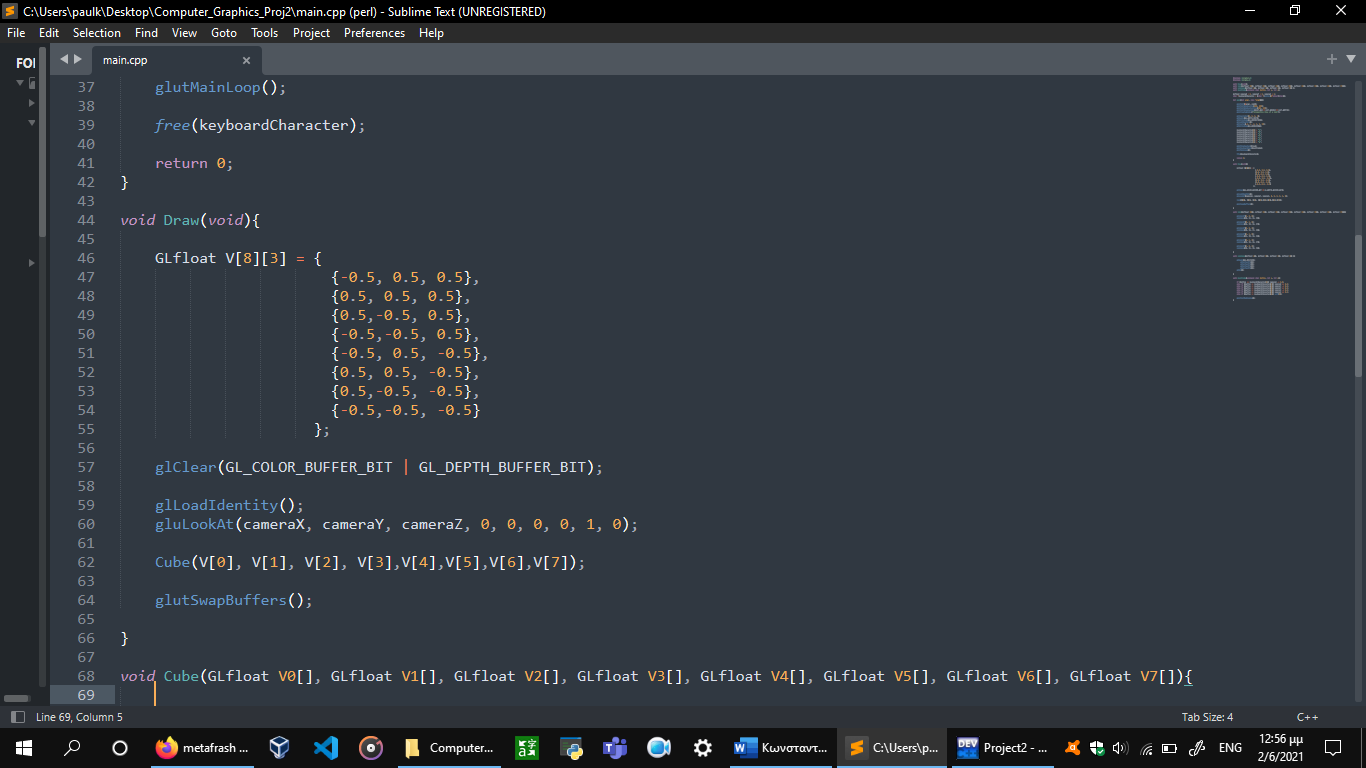


Εικόνα 2

Για να ξεκινήσουμε την σχεδίαση σε ένα «καθαρό» παράθυρο, χρησιμοποιούμε την συνάρτηση glClear(GL\_COLOR\_BUFFER\_BIT | GL\_DEPTH\_BUFFER\_BIT), η οποία «καθαρίζει» τους καταχωρητές χρώματος και βάθους. Ύστερα, με τη συνάρτηση glLoadIdentity(), φορτώνουμε τον πίνακα, στον οποίο θα αποθηκευτούν οι διάφοροι μετασχηματισμοί (αρχικά ο πίνακας είναι συνήθως μοναδιαίος).

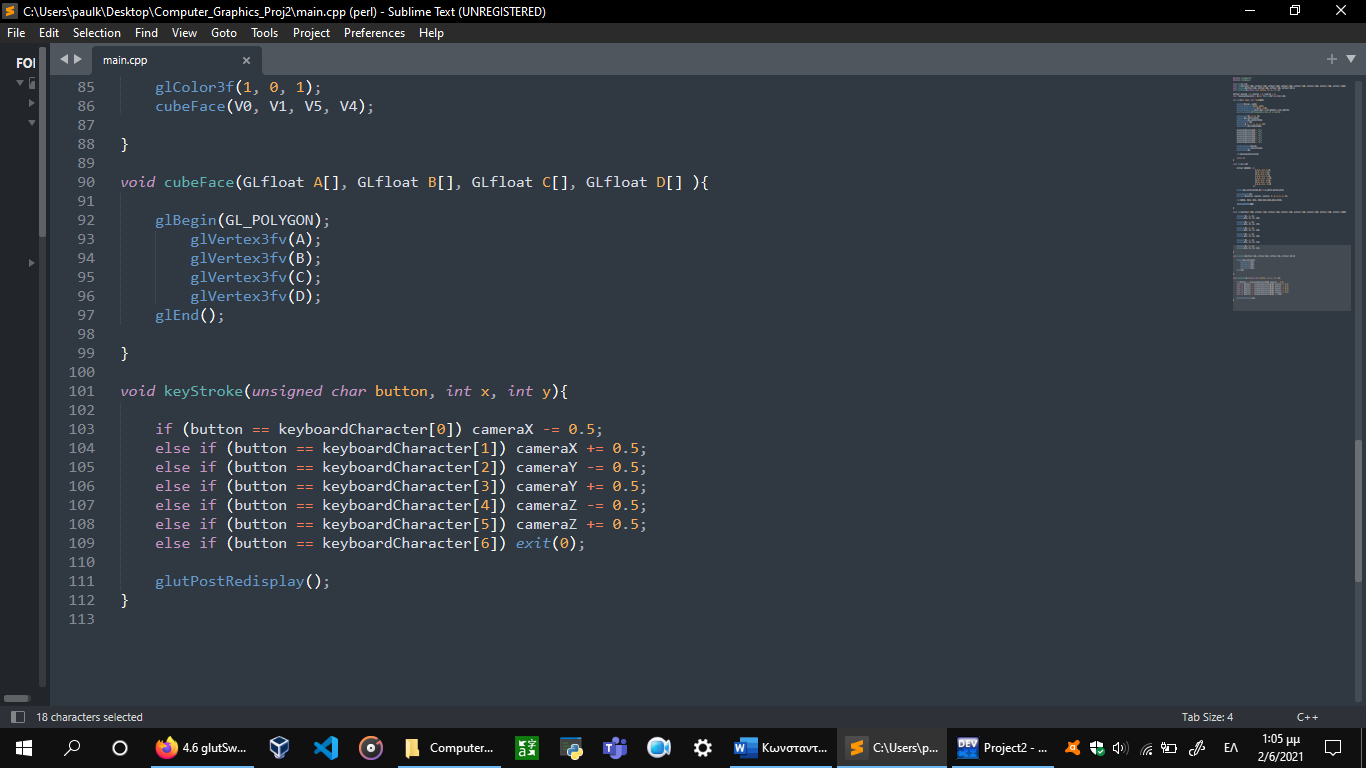
Στην γραμμή 60, με την gluLookAt(cameraX, cameraY, cameraZ, 0, 0, 0, 0, 1, 0), ορίζουμε την αρχή του πλαισίου αναφοράς θέασης ως προς τη θέση των παγκόσμιων συντεταγμένων, δηλαδή ορίζουμε τη θέση της κάμερας μέσα στην σκηνή ως σημείο με συντεταγμένες: Pc = (cameraX, cameraY, cameraZ). Τα επόμενα τρία ορίσματα είναι οι συντεταγμένες της θέσης αναφοράς, δηλαδή η θέση του κύβου στους άξονες και οι τελευταίες τρεις το ανοδικό διάνυσμα, δηλαδή υποδεικνύει ποιός άξονας από τους x, y, z, είναι το επάνω μέρος του κύβου, καθώς δεν μπορούμε να προσδιορίσουμε τον τρόπο θέασης ενός αντικειμένου παρέχοντας μόνο τις συντεταγμένες της κάμερας και του αντικειμένου. Όπως θα έχετε παρατηρήσει, η καθολική μεταβλητή cameraZ αρχικοποιείται στην τιμή 4, γιατί αν της είχε δοθεί μηδενική τιμή, τότε δεν θα μπορούσαμε να δούμε τον κύβο, καθώς κάμερα και κύβος θα βρισκόταν στο ίδιο σημείο.

Ύστερα, καλούμε τη συνάρτηση Cube για τον σχεδιασμό και την τοποθέτηση του κύβου στους άξονες (γραμμή 62). Τέλος, επειδή χρησιμοποιούμε διπλό καταχωρητή για τη σχεδίαση της σκηνής με το κύβο, η συνάρτηση glutSwapBuffers() όταν τελειώσει η σχεδίαση και τοποθέτηση του κύβου στην σκηνή, εναλλάσει τους καταχωρητές για να δούμε το αποτέλεσμα στην οθόνη μας.



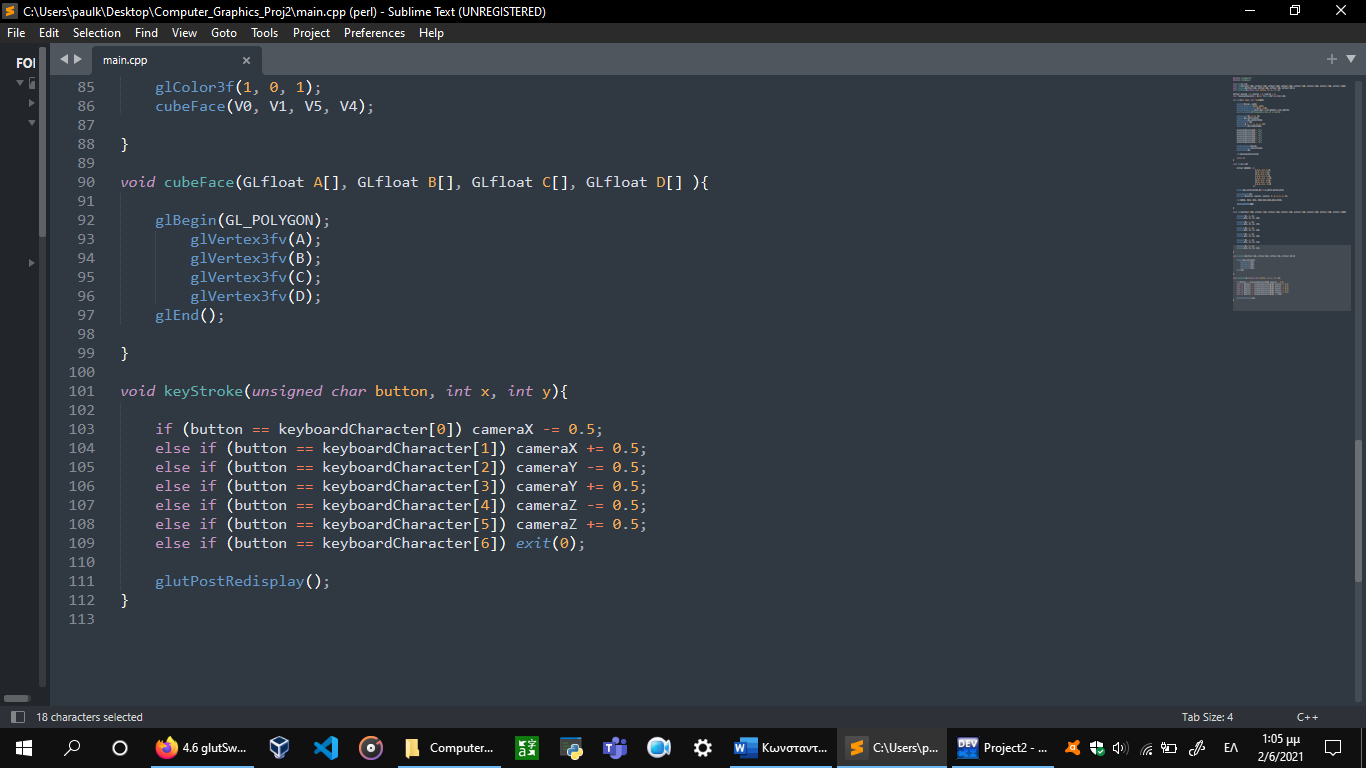
Εικόνα 3: Η συνάρτηση Draw

* Η συνάρτηση **cubeFace()** χρησιμοποιείται για τον σχεδιασμό των έξι πλευρών του κύβου με βάση τις συντεταγμένες που θα ορίσει ο χρήστης ως ορίσματα (σχεδιάζεται μία πλευρά ανά κλήση της συνάρτησης), (εικόνα 4).



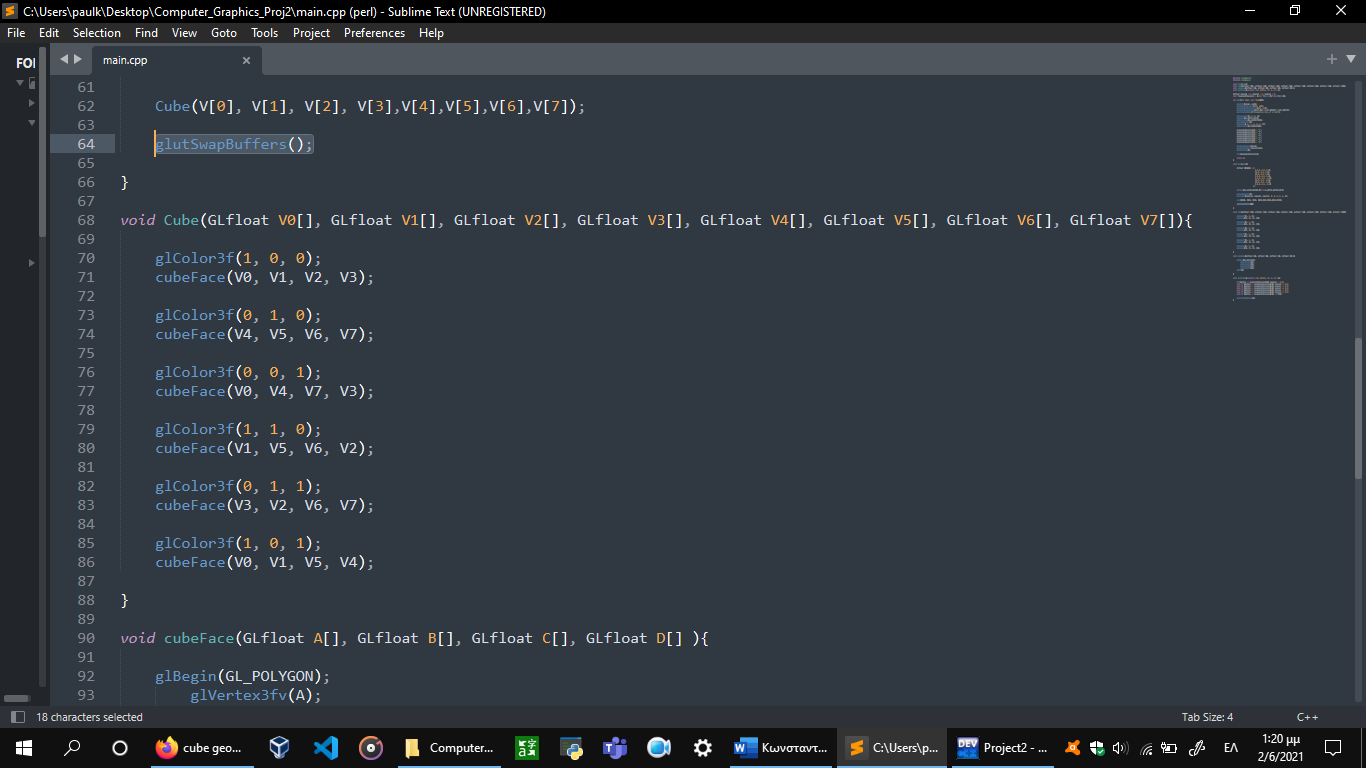
Εικόνα 4: Η συνάρτηση cubeFace

* Η συνάρτηση **keyStroke()**, χρησιμοποιείται για να μεταβάλλει την θέση της κάμερας μέσα στην σκηνή. Όπως αναφέρθηκε και προηγουμένως, ο χρήστης μεσω των κουμπιών του πληκτρολογίουν: ‘x’, ‘y’, ‘z’, ‘X’, ‘Y’, και ‘Z’, να πειραματιστεί με την οπτική γωνία της κάμερας, δηλαδή να αυξομειώσει την θέση της. Με το πάτημα του κουμπιού ‘q’, ο χρήστης μπορεί να τερματίσει το πρόγραμμα.



Εικόνα 5: Η συνάρτηση keyStroke

* Η συνάρτηση Cube(), είναι υπεύθυνη για τον χρωματισμό των πλευρών του κύβου. Πιο ειδικά, με την glColor3f(), επιλέγεται ένα ξεχωριστό χρώμα ανά πλευρά και με την κλήση της cubeFace(), σχεδιάζεται η πλευρά στο παράθυρο (εικόνα).



Εικόνα 6: Η συνάρτηση Cube

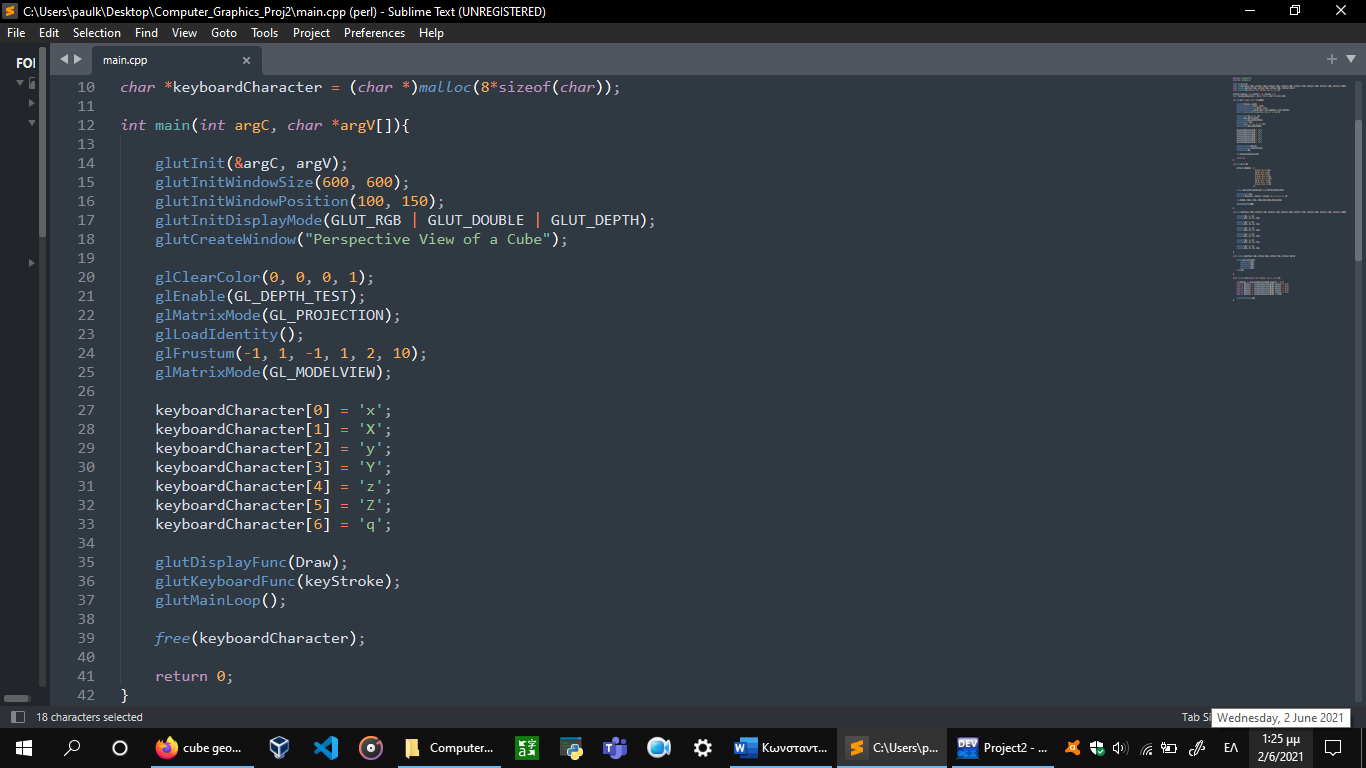
* Στη συνάρτηση main, ορίζουμε το μέγεθος του παραθύρου και τη θέση του στην οθόνη του υπολογιστή (γραμμές 15, 16, εικόνα 7). Στη συνέχεια, ορίζουμε τον τρόπο προβολής του παραθύρου, αρχικοποιώντας τον χρωματισμό σε RGB, διπλό καταχωρητή παραθύρου και τον καταχωρητή βάθους (γραμμή 17) και δημιουργούμε το παράθυρο (γραμμή 18). Με την εντολή glClearColor(0, 0, 0, 1), ορίζουμε το φόντο του παραθύρου στο μαύρο. Έπειτα, στις γραμμές 22-23 φορτώνουμε τον πίνακα μετασχηματισμού προβολής.

Μέσω της glFrustum(), ορίζουμε μια προοπτική προβολή που έχει ένα συμμετρικό κωνικό κόλουρο θέασης1. Οι πρώτες τέσσερις παράμετροι θέτουν τις συντεταγμένες του παραθύρου αποκοπής πάνω στο κοντινό επίπεδο και οι δύο τελευταίες, τις αποστάσειςαπό την αρχή των συντεταγμένων προς τα κοντινά και μακρινά επίπεδα αποκοπής, κατά μήκος του αρνητικού άξονα Ζ.Ο μονοδιάστατος πίνακας

*1Μπορείτε να διαβάσετε αναλυτικότερα από το βιβλίο στις σελίδες 358-365*

keyboardCharacter, περιέχει τα δεσμευμένα κουμπιά του πληκτρολογίου που θα χρησιμοποιηθούν για την μετακίνηση της «κάμερας», όπως αναφέρθηκε προηγουμένως.

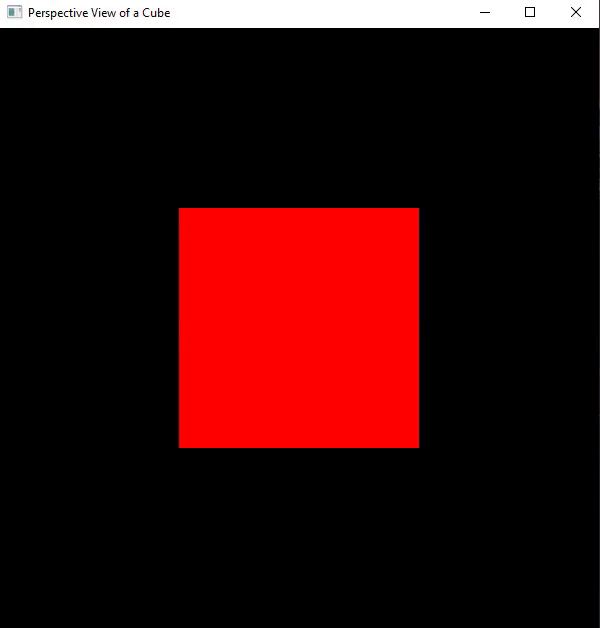
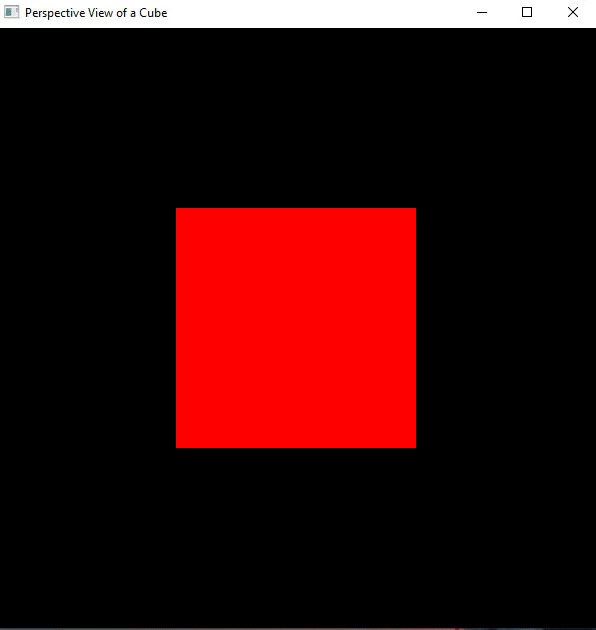
Τέλος, (γραμμές 35-39), καλούμε τη συνάρτηση Draw για τη σχεδίαση της σκηνής με τον κύβο και ελέγχουμε αν πατήθηκε κάποιο από τα κουμπιά του πληκρολογίου που είναι υπεύθυνα για την μετακίνηση της οπτική γωνίας του παραθύρου θέασης. Όταν τελειώσει η εκτέλεση του προγράμματος, απελευθερώνεται ο χώρος που δεσμεύτηκε από τον δυναμικό πίνακα.



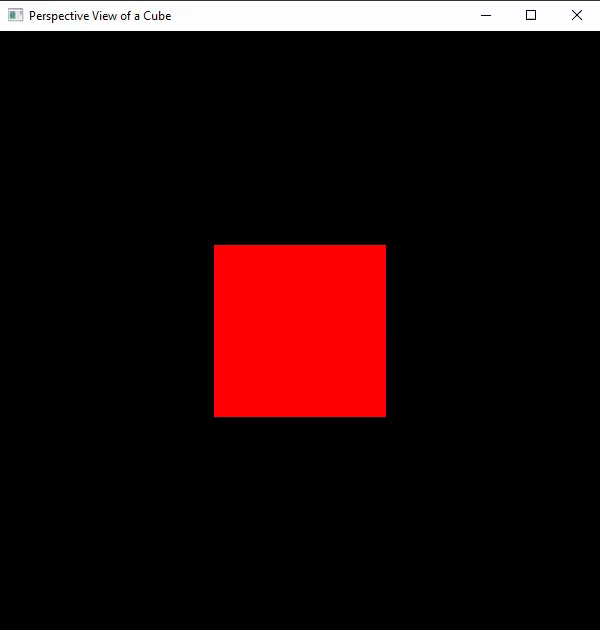
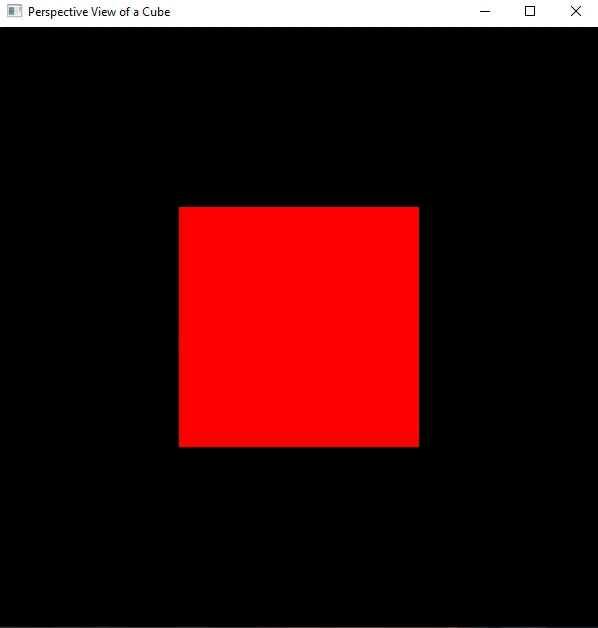
Εικόνα 7: Η συνάρτηση main

**1.4 Αναπαράσταση του Αποτελέσματος του Αλγορίθμου**

Παρακάτω φαίνεται ο τρόπος με τον οποίο η οπτική γωνία θέασης μεταβάλλεται, με την χρήση του πληκτρολογίου για την μετακίνηση της κάμερας. Παρατηρούμε ότι αν απομακρυνθεί αρκετά η κάμερα από τον κύβο, τότε το σχήμα αποκόπτεται από την σκηνή (Διαδραστική Εικόνα 11).



Διαδραστική Εικόνα 8: Μετακίνηση της κάμερας ως προς Διαδραστική Εικόνα 9: Μετακίνηση της κάμερας ως προς τον άξονα x . τον άξονα y (angle).



Διαδραστική Εικόνα 10: Μετακίνηση της κάμερας ως προς τον άξονα z (near-far). Διαδραστική Εικόνα 11: Τριδιάστατη Αποκοπή.

**1.5 Συμπεράσματα**

Το πρόγραμμα που παρουσιάστηκε προηγουμένως, δύναται να χρησιμποποιήσει ως σκηνή οποιοδήποτε στερεό, αρκεί ο προγραμματιστής να ορίσει σωστά τις συντεταγμένες των κορυφών του μέσα στη συνάρτηση Draw() και να προσέξει αυτές να ορίζονται μέσα στα όρια του παραθύρου θέασης. Με τα κουμπιά του πληκτρολογίου, ‘x’, ‘y’, ‘z’, ‘X’, ’Y’, ’Z’, ο χρήστης μπορεί να μεταβάλλει τη θέση της κάμερας μέσα στη σκηνή και κατά συνέπεια, την οπτική γωνία της προοπτικής προβολής.

**1.6 Επίλογος**

Εν κατακλέιδι, από τα είδη προβολών που υπάρχουν, ιδιαίτερο ενδιαφέρον παρουσιάζουν: η παράλληλη (ή ορθογραφική) προβολή και η προοπτική προβολή. Σε αντίθεση με την παράλληλη προβολή, οι σκηνές (ή προβολές) που δημιουργούνται με την προοπτική, παρουσιάζουν μεγαλύτερη ρεαλιστικότητα διότι καθώς ένα αντικείμενο απομακρύνεται από τον θεατή, εμφανίζεται μικρότερο στην σκηνή. Όταν ο θεατής απομακρυνθεί σημαντικά από το αντικείμενο, τότε αυτό αποκόπτεται, δηλαδή δεν προβάλλεται στην σκηνή.

*[ Ο κώδικας του που αναπτύχθηκε: ]*

**1.7 Βιβλιογραφία**

1. Κ. Μουστάκας, Ι. Παλιόκας, Α. Τσακίρης, Δ.Τζοβάρας, “Γραφικά και Εικονική Πραγματικότητα”, 2015.
2. Hearn, Baker, Επιστημονική επιμέλεια: Παναγιώτης Μποζάνης, καθηγητής Πανεπιστημίου Θεσσαλίας, “Γραφικά Υπολογιστών με OpenGL 3η Βελτιωμένη Έκδοση”.

**1.8 Ευρετήριο**

προοπτική προβολή 5,9,12

σκηνή 3,6,7,8,10,11,12,13

κάμερα 3,5,6,7,8,10,11,12

κύβος 3,7

θέαση 3,6,7,9,10,11,12

μετακίνηση 5,10,11,12