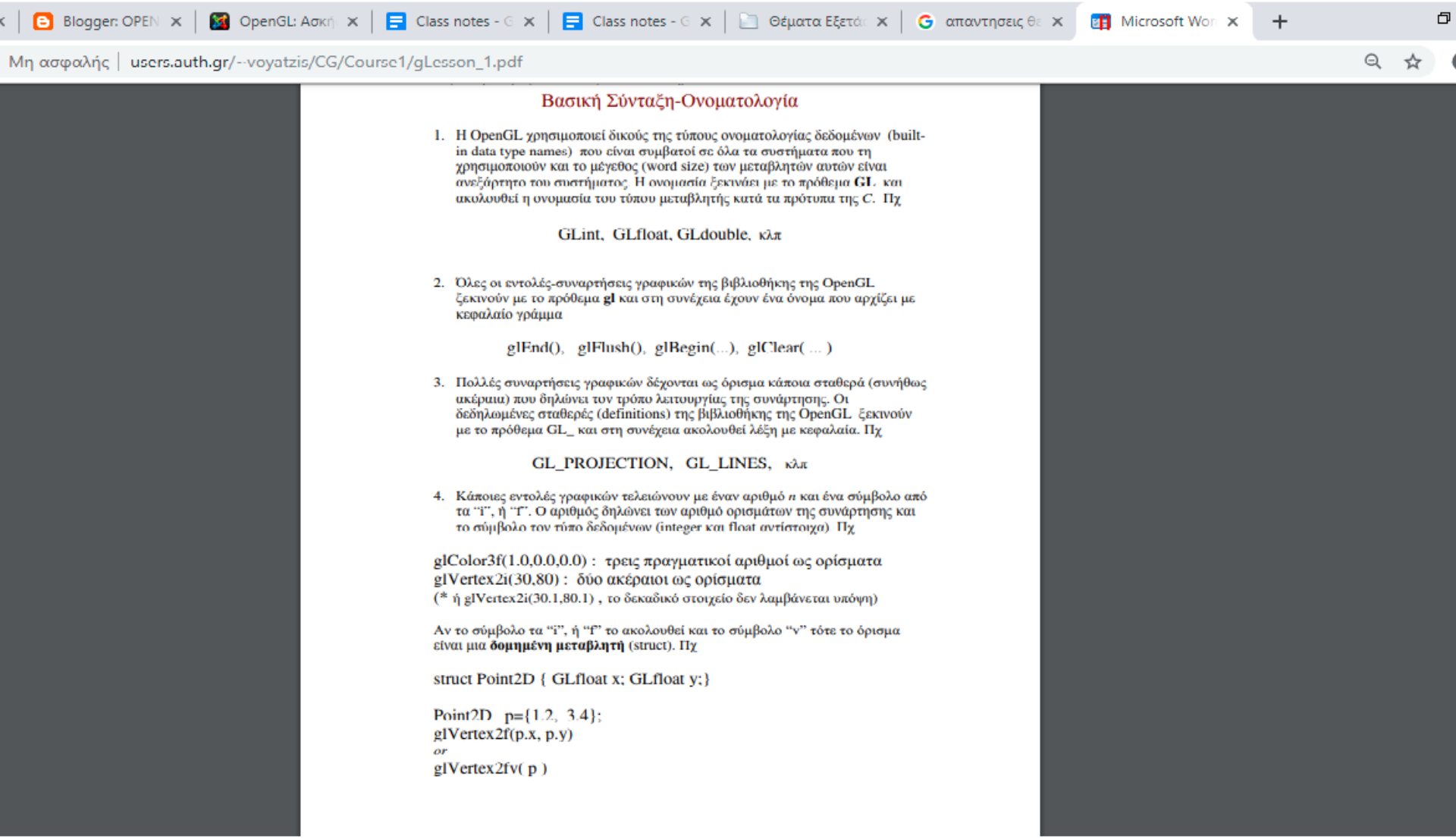
Ποια η λειτουργία των παρακάτω εντολών στην OpenGL και τι παραμέτρους δέχονται; Να δοθεί παράδειγμα.α) glutInitWindowPosition β) glutInitWindowSize γ) glutCreateWindow

glutInitWindowPosition και _ Το glutInitWindowSize ορίζει την *αρχική θέση* και το *μέγεθος του παραθύρου* αντίστοιχα.

Εξετάστε την ισχύ των παρακάτω προτάσεων δηλώνοντας ΣΩΣΤΟ ή ΛΑΘΟΣ και εξηγώντας την επιλογή σας (Όπου κρίνετε απαραίτητο, παραθέστε παράδειγμα)

- Τα επικρατέστερα Λειτουργικά Συστήματα (Windows, Unix, Linux, Mac OS) υποστηρίζουν την OpenGL. ΣΩΣΤΟβ)
- Η OpenGL μπορεί αποκλειστικά να κληθεί (is callable) από τις γλώσσες προγραμματισμού C / C++ (δηλαδή υπάρχει μοναδικό language binding). ΛΑΘΟΣ
- Το μοναδικό περιβάλλον ανάπτυξης προγραμμάτων OpenGL είναι το DEV C++. ΛΑΘΟΣ
- Η OpenGL περιέχει εντολές επιλογής (τύπου If ... else). ΣΩΣΤΟ
- Οι εντολές της OpenGL ξεκινούν με το πρόθεμα gl. ΣΩΣΤΟ

Η OpenGL χρησιμοποιεί μια απλή, βασική, μορφή ονοματολογίας για τις εντολές της. Αναφέρατε ποια είναι αυτή και παραθέστε παράδειγμα



Βασική Σύνταξη-Ονοματολογία

1. Η OpenGL χρησιμοποιεί δικούς της τύπους ονοματολογίας δεδομένων (built-in data type names) που είναι συμβατοί σε όλα τα συστήματα που τη χρησιμοποιούν και το μέγεθος (word size) των μεταβλητών αυτών είναι *ανεξάρτητο του συστήματος*. Η ονομασία ξεκινάει με το πρόθεμα **GL** και ακολουθεί η ονομασία του τύπου μεταβλητής κατά τα πρότυπα της *C*. Πχ

   GLint, GLfloat, GLdouble, κλπ

2. Όλες οι εντολές-συναρτήσεις γραφικών της βιβλιοθήκης της OpenGL ξεκινούν με το πρόθεμα **gl** και στη συνέχεια έχουν ένα όνομα που αρχίζει με κεφαλαίο γράμμα

   glEnd(), glFlush(), glBegin(...), glClear(...)

3. Πολλές συναρτήσεις γραφικών δέχονται ως όρισμα κάποια σταθερά (συνήθως ακέραια) που δηλώνει τον τρόπο λειτουργίας της συνάρτησης. Οι δεδηλωμένες σταθερές (definitions) της βιβλιοθήκης της OpenGL ξεκινούν με το πρόθεμα GL_ και στη συνέχεια ακολουθεί λέξη με κεφαλαία. Πχ

   GL_PROJECTION, GL_LINES, κλπ

4. Κάποιες εντολές γραφικών τελειώνουν με έναν αριθμό *n* και ένα σύμβολο από τα "i", ή "f". Ο αριθμός δηλώνει των αριθμό ορισμάτων της συνάρτησης και το σύμβολο τον τύπο δεδομένων (integer και float αντίστοιχα) Πχ

glColor3f(1.0,0.0,0.0) : τρεις πραγματικοί αριθμοί ως ορίσματα
glVertex2i(30,80) : δύο ακέραιοι ως ορίσματα
(* ή glVertex2i(30.1,80.1) , το δεκαδικό στοιχείο δεν λαμβάνεται υπόψη)

Αν το σύμβολο τα "i", ή "f" το ακολουθεί και το σύμβολο "v" τότε το όρισμα είναι μια **δομημένη μεταβλητή** (struct). Πχ

```
struct Point2D { GLfloat x; GLfloat y;}

Point2D  p={1.2, 3.4};
glVertex2f(p.x, p.y)
or
glVertex2fv( p )
```

Περιγράψτε τις βασικές εντολέςτης OpenGL που περιέχουν εντολές σχεδίασης, γραφικών και απόδοσης

Βασικές εντολές της GLUT

Οι εντολές της GLUT αρχίζουν με το πρόθεμα **glut** και χρησιμοποιούνται για την δημιουργία και διαχείριση παραθύρων γραφικών μέσα από ένα **προγραμματιστικό περιβάλλον κονσόλας**.

glutInit(&argc, argv) :
Αρχικοποιεί τον τρόπο αλληλεπίδρασης μεταξύ του περιβάλλοντος της εφαρμογής με την βιβλιοθήκη της OpenGL. Η συνάρτηση δέχεται υποχρεωτικά τα βασικά ορίσματα του προγράμματος με τα οποία μπορεί ο χρήστης να ορίσει εσωτερικές διαδικασίες και ελέγχους.

glutInitDisplayMode(*mode*)
Καθορίζει το τρόπο παρουσίασης των γραφικών (χρώμα, buffering, απόκρυψη αντικειμένων κλπ). Οι διάφορες παράμετροι *mode* χωρίζονται μεταξύ τους με τον τελεστή or (|).
πχ glutInitDisplayMode(GLUT_SINGLE|GLUT_RGB) = μονό buffer , χρώμα RGB.

glutInitWindowPosition(pos_x, pos_y):
Αρχική θέση (σε pixels) του παραθύρου γραφικών στην οθόνη. (default: system's control)

glutInitWindowSize(size_x, size_y):
Μέγεθος παραθύρου σε pixels (default=300x300)

glutCreateWindow("*title*"):
Δημιουργεί και παρουσιάζει το παράθυρο (virtually) λαμβάνοντας υπόψη τα παραπάνω στοιχεία θέσης και μεγέθους. Η παρουσίαση του παραθύρου στην οθόνη γίνεται με την ολοκλήρωση του απαραίτητου κύκλου εντολών.

glutDisplayFunc(*user's_function*)
Ορίζει (registers) τη συνάρτηση του χρήστη που εκτελεί τη σχεδίαση των γραφικών (*display call back function*). Η συνάρτηση αυτή εκτελείται κάθε φορά που το σύστημα θεωρεί ότι το παράθυρο πρέπει να ανανεωθεί (refresh).

glutMainLoop( )
Εκτελεί τις εντολές γραφικών : παρουσιάζει το παράθυρο γραφικών στη οθόνη και εκτελεί τον κώδικα σχεδίασης (*display call back functions*). Ένα πρόγραμμα της OpenGL αποτελείται από ένα σύνολο και μια οργάνωση από *display call back functions*. Επίσης με την glutMainLoop το πρόγραμμα εισέρχεται σε κατάσταση *βρόχου επεξεργασίας* (glut processing loop) με συνεχή έλεγχο των «events» (ενέργειες αλληλεπίδρασης με τις μονάδες εισόδου).

Αναφέρετε την λειτουργία της glutMainLoop() στην OpenGL. Σε ποιο σημείο πρέπει να καλείται;

glutMainLoop( )
Εκτελεί τις εντολές γραφικών : παρουσιάζει το παράθυρο γραφικών στη οθόνη και εκτελεί τον κώδικα σχεδίασης (*display call back functions*). Ένα πρόγραμμα της OpenGL αποτελείται από ένα σύνολο και μια οργάνωση από *display call back functions*. Επίσης με την glutMainLoop το πρόγραμμα εισέρχεται σε κατάσταση *βρόγχου επεξεργασίας* (glut processing loop) με συνεχή έλεγχο των «events» (ενέργειες αλληλεπίδρασης με τις μονάδες εισόδου).

Περιγράψτε τη βασική δομή μιας εφαρμογής φτιαγμένης σε OpenGL.

```
#include
void display()
{ glClearColor(1,1,1,1);
 glClear(GL_COLOR_BUFFER_BIT);
 glBegin(GL_LINES);
 glColor3f(1,0,0);
 glVertex2i(20,20);
 glVertex2i(40,40);
 glEnd();
 glFlush(); }
 int main(int argc, char** argv)
{ glutInit(&argc,argv);
 glutInitWindowPosition(50,50);
glutInitWindowSize(640,480);
glutInitDisplayMode(GLUT_SINGLE|GLUT_RGB);
 glutCreateWindow("A sample OpenGL application");
glMatrixMode(GL_PROJECTION);
 gluOrtho2D(0,50,0,50);

glutDisplayFunc(display);
 glutMainLoop(); return 0; }
```

Περιγράψτε τρεις βασικές βιβλιοθήκες εντολών της OpenGL που περιέχουν εντολές σχεδίασης, γραφικών και απόδοσης. Σε ποιο σημείο του προγράμματος δηλώνονται;

- α) Βασική βιβλιοθήκη (OpenGL core library): Η βασική βιβλιοθήκη της OpenGL περιέχει τις κύριες εντολές σχεδίασης. Ολες οι εντολές της βιβλιοθήκης αυτής διακρίνονται από το πρόθεμα gl . Πολλές από τις συναρτήσεις της δέχονται προκαθορισμένα ορίσματα (συμβολικές σταθερές) τα οποία έχουν οριστεί στη βιβλιοθήκη και αντιστοιχούν σε διάφορες παραμέτρους ή καταστάσεις λειτουργίας. Κατά σύμβαση, οι σταθερές αυτές ξεκινούν με το πρόθεμα GL_ .
- β) OpenGL Utility Library (GLU): Περιλαμβάνει συναρτήσεις που εκτελούν σύνθετους αλγορίθμους όπως π.χ. τον καθορισμό μητρώων προβολής και το σχηματισμό σύνθετων καμπυλών και επιφανειών. Κάθε υλοποίηση της OpenGL εμπεριέχει τη βιβλιοθήκη GLU. Όλες οι εντολές της βιβλιοθήκης GLU ξεκινούν με το πρόθεμα glu.
- γ) OpenGL Utility Toolkit (GLUT): Όπως αναφέραμε το πρότυπο της OpenGL είναι ανεξάρτητο πλατφόρμας. Ωστόσο μια αυτονόητη απαίτηση ενός προγραμματιστή είναι να έχει τη δυνατότητα να δει το αποτέλεσμα των προγραμμάτων του αλλά και να έχει δηλαδή τη δυνατότητα αλληλεπίδρασης με αυτό. Χρειάζεται δηλαδή εντολές εισόδου-εξόδου. Ωστόσο οι εντολές αλληλεπίδρασης δεν είναι ανεξάρτητες πλατφόρμας και για το λόγο αυτό οι προγραμματιστές σε OpenGL χρησιμοποιούν μία ακόμη βιβλιοθήκη που τους προσφέρει εντολές εισόδουεξόδου. Μία από τις βιβλιοθήκες που προσφέρει τη λειτουργικότητα αυτή είναι το OpenGL Utility Toolkit (GLUT). Η βιβλιοθήκη αυτή περιλαμβάνει εντολές απεικόνισης παραθύρων στην οθόνη, δημιουργίας menus, διαχείρισης γεγονότων κλπ. Όλες οι εντολές της ξεκινούν με το πρόθεμα glut.

Ποιες βασικές λειτουργίες υποστηρίζει η OPENGL;

- Λειτουργίες της GLU:
  Αρχικοποίηση Χειρισμός εικόνων για χρήση σε υφή Μετασχηματισμός σημείων 2.4 Tessellating πολύγωνων Χρήση λειτουργιών επανάκλησης Χρησιμοποιώντας ψηφιδωτά(Tessellation) αντικείμενα Καθορισμός επανακλήσεων Καθορισμός του πολύγωνου ως ψηφιδωτό(Tessellated) Απόδοση απλών επιφάνειων Χρησιμοποιώντας NURBS καμπύλες και επιφάνειες Χειρισμός σφαλμάτων

Τι είναι η OpenGL

- Η OpenGL είναι ένα σύνολο εντολών (Application Programming Interface – API) που μας επιτρέπει την δημιουργία τριδιάστατων γραφικών. Δεν είναι γλώσσα προγραμματισμού αλλά μπορεί να χρησιμοποιηθεί με μια πληθώρα γλωσσών προγραμματισμού (C, C++, Java και άλλες) και σε μια πληθώρα λειτουργικών συστημάτων (Windows, Unix, Linux, Mac OS). Προσφέρει πάνω από 300 εντολές για δημιουργία γραφικών και δρα σαν ένα ενδιάμεσο στρώμα ανάμεσα στην εφαρμογή και στη κάρτα γραφικών που θα αναλάβει τα απεικονίσει τα γραφικά στην οθόνη, κρύβοντας λεπτομέρειες υλοποίησης του υλικού και των οδηγών του.

TI είναι το OpenGL graphics pipeline

- Ας δούμε σχηματικά τα βήματα που κάνει η OpenGL από την στιγμή που πάρει τις εντολές γραφικών από την εφαρμογή μας μέχρι την στιγμή δημιουργίας της τελικής εικόνας (rendered image). Η OpenGL χρησιμοποιεί μια μηχανή καταστάσεων (state machine) για να επικοινωνεί με την εφαρμογή. Σε αυτή την μηχανή καταστάσεων η OpenGL παραμένει διαρκώς σε μια κατάσταση μέχρι να αλλάξει η εφαρμογή την κατάσταση. Παράδειγμα αν θέσουμε το χρώμα που θα χρησιμοποιεί η OpenGL για να ζωγραφίσει ένα μοντέλο, το χρώμα θα παραμείνει στη μνήμη και θα χρησιμοποιείται μέχρι να το αλλάξουμε ή να κλείσουμε την εφαρμογή. Εφόσον λοιπόν η εφαρμογή καθορίσει το περιβάλλον που θα χρησιμοποιήσει η OpenGL για να απεικονίσει το μοντέλο μας (χρώματα, υφές, πηγές φωτός, κάμερα κλπ), περνάει στο πρώτο στάδιο κατά το οποίο θα μετασχηματίσει και θα φωτίσει (transform and lighting) τα σημεία(vertices) του μοντέλου. Στην συνέχεια η OpenGL περνά στο στάδιο της ψηφιοποίησης το οποίο λαμβάνει όλες τις πληροφορίες και την γεωμετρία από το προηγούμενο στάδιο (του μετασχηματισμού) και παράγει την τελική, ψηφιακή, εικόνα. Η εικόνα αντιγράφεται στην μνήμη του frame buffer και φτάνει έτσι στην οθόνη του υπολογιστή. Αυτή η ακολουθία βημάτων που ακολουθεί η OpenGL για να απεικονίσει το μοντέλο λέγεται graphics pipeline (διασωλήνωση)

Τι είναι το GLUT

- Η OpenGL όπως είπαμε είναι ένας γρήγορος και ευέλικτος τρόπος να επικοινωνούμε με το hardware γραφικών του υπολογιστή χωρίς να ενδιαφερόμαστε για τις λεπτομέρειες υλοποίησης του. Όμως δεν προσφέρει καθόλου λειτουργίες GUI (Graphical User Interface), δηλαδή δεν έχει την δυνατότητα να ανοίξει και να κλείσει παράθυρα στο λειτουργικό σύστημα, να ζωγραφίσει σε αυτά, ούτε να καταλάβει το πάτημα ενός πλήκτρου ή την κίνηση του ποντικιού, ούτε μπορεί να διαβάσει ένα αρχείο από το δίσκο. Αυτό έγινε επί σκοπού, μιας και η OpenGL σχεδιάστηκε να τρέχει σε πολλά λειτουργικά συστήματα τα οποία έχουν το δικό τους τρόπους επικοινωνίας με την οθόνη, το ποντίκι, το δίσκο, το πληκτρολόγιο. Για να γίνει αυτό θα πρέπει να χρησιμοποιήσουμε απευθείας τις εντολές του λειτουργικού μας συστήματος (Win32 εντολές στην περίπτωση των Windows). Εναλλακτικά μπορούμε να χρησιμοποιήσουμε μια από τις έτοιμες βιβλιοθήκες εντολών που υπάρχουν και που θα κάνουν αυτές τις λειτουργίες για εμάς εύκολα. Μια από τις πιο διαδεδομένες βιβλιοθήκες για αυτό το σκοπό είναι το GLUT (OpenGL Utility Toolkit) το οποίο είναι και αυτό σχεδιασμένο να τρέχει σε πολλά λειτουργικά συστήματα.

ποιους τύπους δεδομένων γνωρίζετε

- Τύποι δεδομένων OpenGL Χάριν ομοιομορφίας, και για να μπορέσει να υποστηρίξει διαφορετικές πλατφόρμες, η OpenGL ορίζει μια σειρά από τύπους δεδομένων οι κυριότεροι των οποίων φαίνονται στο παρακάτω πίνακα: Τύπος δεδομένων OpenGL Ορίζεται ως Αντιστοιχία με τύπο δεδομένων στη C Επίθεμα GLbyte Ακέραιος 8-bit signed char b GLshort Ακέραιος 16-bit short s GLint Ακέραιος 32-bit int i GLdouble Κινητής υποδιαστολής 64-bit double d GLubyte Θετικός ακέραιος 8-bit unsigned char ub GLushort Θετικός ακέραιος 16-bit unsigned short us GLuint Θετικός ακέραιος 32-bit unsigned int ui GLchar Χαρακτήρας 8-bit char - GLfloat Κινητής υποδιαστολής 32-bit float f GLboolean

Εξηγείστε την OpenGL ως μηχανή καταστάσεων

- Το περιβάλλον της OpenGL μπορεί να χαρακτηριστεί ως μια μηχανή καταστάσεων. Με τον όρο “μηχανή καταστάσεων” αναφερόμαστε σε ένα περιβάλλον, το οποίο, σε κάθε χρονική στιγμή, λειτουργεί βάσει προκαθορισμένων ιδιοτήτων (attributes) ή αλλιώς μεταβλητών κατάστασης (state variables). Οι μεταβλητές κατάστασης έχουν μια προκαθορισμένη αρχική τιμή η οποία μπορεί να μεταβληθεί κατά την πορεία της εκτέλεσης του κώδικα από τον προγραμματιστή. Επιπλέον, οι αρχικές τιμές τους ή οι τιμές που τους ανατέθηκαν την τελευταία φορά, παραμένουν ενεργές.
  Δεδομένου ότι στην OpenGL οι αλγόριθμοι εκτελούνται επαναληπτικά, είναι σημαντικό ο προγραμματιστής να αρχικοποιεί τις μεταβλητές κατάστασης, όποτε αυτό είναι απαραίτητο, καθώς και να παρακολουθεί τις τιμές τους, ούτως ώστε να παράγει το επιθυμητό αποτέλεσμα σε κάθε κύκλο εκτέλεσης.
  Ορισμένα παραδείγματα μεταβλητών κατάστασης που ορίζονται στην OpenGL είναι: το τρέχον χρώμα σχεδίασης, οι τιμές των μητρώων μετασχηματισμού και προβολής, το πάχος των σχεδιαζόμενων γραμμών, το χρώμα καθαρισμού της οθόνης κ.λ.π.

Πως καθαρίζουμε τον ενταμιευτή χρωματικών τιμών (color buffer) και τον ενταμιευτή τιμών βάθους

- για να καθαρίσουμε λ.χ. τον ενταμιευτή χρωματικών τιμών (color buffer) και τον ενταμιευτή τιμών βάθους (depth buffer) δίνουμε:
  glClearColor(0.0, 0.0, 0.0, 0.0);
  glClearDepth(0.0);
  glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT);
  Η εντολή glClearColor() είναι ίδια με αυτήν του προηγούμενου παραδείγματος. ενώ εντολή glClearDepth() καθορίζει την τιμή την οποία θα δώσουμε σε κάθε pixel του ενταμιευτή βάθους (περισσότερα για τη χρήση ενταμιευτών βάθους στην ενότητα “Καταστολή κρυμμένων επιφανειών”).

πως γίνεται ο Καθορισμός Χρωμάτων

- Με την OpenGL, η περιγραφή του σχήματος ενός αντικειμένου και ο χρωματισμός του είναι 20
ανεξάρτητα. Κάθε φορά που δίνουμε εντολή σχεδιάσης ενός συγκεκριμένου γεωμετρικού σχήματος, το τρέχον επιλεγμένο χρώμα, όντας μεταβλητή κατάστασης, καθορίζει και το χρώμα με το οποίο το σχήμα σχεδιάζεται.
Για να καθορίσουμε ένα χρώμα, χρησιμοποιούμε την εντολή glColor3f(). Η εντολή αυτή παίρνει τρεις παραμέτρους, οι οποίες είναι όλες αριθμοί κινητής υποδιαστολής ή διπλής ακρίβειας (μεταξύ 0.0 και 1.0) ή ακέραιοι, ανάλογα με το ποια από τις παρακάτω τρείς μορφές επιλέγουμε:
void glColor3ub(GLubyte red, GLubyte green, GLubyte blue);
(0≤red,green,blue≤255)
void glColor3f(GLfloat red, GLfloat green, GLfloat blue); (0≤red,green,blue≤1)
void glColor3d(GLdouble red, GLdouble green, GLdouble blue);
(0≤red,green,blue≤1)
Οι παράμετροι red, green και blue αντιστοιχούν στις τιμές των χρωματικών συνιστωσών του χρώματος στο μοντέλο RGB. Π.χ. η εντολή
glColor3f(1,0,0);
επιστρέφει το πιο έντονο κόκκινο που μπορεί να αποδώσει το σύστημα. Στον πίνακα 3 ορίζονται ορισμένα βασικά χρώματα.

πως γίνεται Καθορισμός κορυφών

- Στην OpenGL, όλα τα γεωμετρικά σχήματα περιγράφονται δηλώνοντας τις κορυφές τους. Για τον καθορισμό μιας κορυφής χρησιμοποιούμε την εντολή glVertex*.
  void glVertex{234}{sifd}[v](TYPE coords);
  Η εντολή αυτή καθορίζει μία κορυφή, η οποία θα χρησιμοποιηθεί για την περιγραφή ενός γεωμετρικού σχήματος. Μπορούμε να δώσουμε μέχρι τέσσερις συντεταγμένες (x, y, z, w) για μία συγκεκριμένη κορυφή ή και μέχρι δύο (x, y), χρησιμοποιώντας την κατάλληλη εκδοχή της εντολής. Η εντολή glVertex*() πρέπει να 21
  εκτελείται μεταξύ των εντολών glBegin() και glEnd(), όπως θα δούμε στη συνέχεια . Παρακάτω βλέπουμε μερικά παραδείγματα όπου χρησιμοποιείται η glVertex*:
  glVertex2s(2,3); // Δήλωση σημείου με συντεταγμένες (x,y)=(2,3)
  glVertex3d(0,0,3.14); // Δήλωση σημείου με συντεταγμένες (x,y,z)=(0,0,3.14)
  glVertex4f(2.3, 1.0, -2.2, 2.0);
  GLdouble dvect[3] = {5,9,1992};
  glVertex3dv(dvect); //Δήλωση σημείου που οι τιμές του βρίσκονται στο μητρώο

Περιγράψτε τη διαδικασία δημιουργίας Τριγώνων

- Ανάλογα με το όρισμα της εντολής glBegin, διακρίνουμε τις εξής διαφοροποιήσεις στη σχεδίαση των τριγώνων.
  GL_TRIANGLES
  Σχεδιάζει μια σειρά από σαφώς διακεκριμένα τρίγωνα. Για διατεταγμένο σύνολο κορυφών σχεδιάζονται τα τρίγωνα με κορυφές (), 0121,,,...,n−vvvv012v,v,v()345v,v,v και ούτω καθεξής. Αν το δεν είναι ακέραιο πολλαπλάσιο του 3, η τελευταία ή οι δύο τελευταίες κορυφές παραλείπονται. n 35
  GL_TRIANGLE_STRIP
  Σχεδιάζει μια αλληλουχία τριγώνων. Διαδοχικά τρίγωνα έχουν μία κοινή πλευρά. Δίνοντας ένα διατεταγμένο σύνολο κορυφών σχεδιάζονται τα τρίγωνα με κορυφές , , και ούτω καθεξής. Η συγκεκριμένη σειρά εμφάνισης των κορυφών εξασφαλίζει ότι οι δήλωσεις των κορυφών των τριγώνων έχουν τον ίδιο προσανατολισμό. (αριστερόστροφο ή δεξιόστροφο) και παίζει ρόλο στον προσανατολισμό της επιφανείας των πολυγώνων, όπως θα δούμε στην ενότητα “Όψεις πολυγώνων”. Το n πρέπει να είναι τουλάχιστο ίσο με 3, ειδάλλως δε θα σχεδιαστεί τίποτα. (011,,...,n−v vv)
)
  ()012v,v,v()213,,vvv(423v,v,v
  GL_TRIANGLE_FAN
  Σχεδιάζει τρίγωνα που έχουν το πρώτο σημείο κοινό και διαδοχικά τρίγωνα έχουν μία κοινή πλευρά. Επομένως τα τρίγωνα σχηματίζονται από τις κορυφές ()012v,v,v, ()023v,v,v, ()034v,v,v και ούτω καθεξής.

Ποιες εντολές χρησιμοποιούμε για τις εξής λειτουργίες της OpenGL: α. translate β. rotate γ. scale ή stretch;

- a.Μετακίνηση
- Η μετακίνηση (translation) ενός αντικειμένου γίνεται με την εντολή glTranslatef(GLfloat X, GLfloat Y, GLfloat Z);
- Η εντολή αυτή παίρνει ως παραμέτρους την μετακίνηση που επιθυμούμε κατά άξονα, κατασκευάζει ένα μετασχηματισμό μετακίνησης T (όπως είδαμε στην θεωρία) και τον εφαρμόζει (πολ/ζει) με τον μετασχηματισμό Μοντέλου-Κάμερας.
- b.Περιστροφή
- Η περιστροφή (rotation) ενός αντικειμένου γίνεται με την εντολή glRotatef(GLfloat angle, GLfloat x, GLfloat y, GLfloat z);
- Η εντολή αυτή κατασκευάζει ένα μετασχηματισμό περιστροφής γύρω από το διάνυσμα που ορίζεται από την τριάδα [x, y, z], κατά γωνία angle (μετριέται σε μοίρες 0-360) με φορά αντίθετη των δεικτών του ρολογιού. Αν μας ενδιαφέρει απλά να περιστρέψουμε το αντικείμενο γύρω από ένα άξονα (X, Y ή Z) μπορούμε να ορίσουμε το διάνυσμα ως [1, 0, 0] για το περιστροφή γύρω από τον X, ως [0, 1, 0] για το περιστροφή γύρω από τον Y και ως [0, 0, 1] για το περιστροφή γύρω από τον Z.
- c. Κλίμακα
- Η μεγέθυνση/σμίκρυνση (scaling) ενός αντικειμένου γίνεται με την εντολή glScalef(GLfloat x, GLfloat y, GLfloat z); όπου x, y, z η κλίμακα του αντικειμένου ανά άξονα

Ποια είναι η λειτουργία της glClear() και ποια της glClearColor3f() (στην OpenGL);

- Η εντολή glClearColor() καθορίζει το χρώμα που χρησιμοποιείται κάθε φορά που εκτελείται εντολή καθαρισμού της οθόνης. Στην OpenGL το χρώμα του φόντου είναι μία μεταβλητή κατάστασης, η οποία διατηρεί την τιμή που της ανατέθηκε την τελευταία φορά. Το χρώμα καθορίζεται από τα βάρη του στο χρωματικό μοντέλο RGB. Στο συγκεκριμένο παράδειγμα, χρησιμοποιείται το λευκό χρώμα.
- • Η εντολή glClear() καθαρίζει ενταμιευτές (buffers), συγκεκριμένες περιοχές μνήμης του συστήματος γραφικών (frame buffer). Η μηχανή γραφικών της OpenGL ορίζει ορισμένες κατηγορίες ενταμιευτών. Με την σταθερά GL_COLOR_BUFFER_BIT δίνουμε εντολή καθαρισμού του ενταμιευτή χρωματικών τιμών (colour buffer). Αυτή περιέχει τις χρωματικές τιμές των pixels που απεικονίζονται (ή πρόκειται να απεικονιστούν) στην οθόνη.
- Με την εντολή glColor3f(float r, float g, float b) ορίζουμε το τρέχον χρώμα σχεδιάσης. Πρόκειται για μία ακόμη μεταβλητή κατάστασης που καθορίζει το χρώμα που χρησιμοποιείται για τη σχεδίαση γραφικών. Στην εντολή περνάμε ως ορίσματα τις κανονικοποιημένες ως προς τη μονάδα τιμές των συνιστωσών του κόκκινου, πράσινου και μπλέ χρώματος. Στο παράδειγμα επιλέγουμε ως χρώμα
- σχεδιάσης το κόκκινο.

δΗΜΙΟΥΡΓΉΣΤΕ ΟΡΘΟΓΩΝΙΑ ΜΕ OPENGL

- Παράδειγμα: Σχεδιάση ορθογωνίων

```
#include <glut.h>
void display()
{
glClearColor(1,1,1,0);
glClear(GL_COLOR_BUFFER_BIT);
glColor3f(1,0,0);
glRectf(0,0,15,10);
glColor3f(0,1,0);
glRectf(20,0,30,10);
glColor3f(0,0,1);
glRectf(12.5,15,25.5,25);
glFlush();
}
int main(int argc, char** argv)
{
glutInit(&argc,argv);
glutInitWindowPosition(50,50);
glutInitWindowSize(640,480);
glutInitDisplayMode(GLUT_SINGLE|GLUT_RGB);
glutCreateWindow("Drawing rectangles");
glMatrixMode(GL_PROJECTION);
gluOrtho2D(-5,35,-5,30);
glutDisplayFunc(display);
glutMainLoop();
return 0;
}
```