

## Εισαγωγή στην OpenGL

### E.1 Τι είναι η OpenGL;

Ένας νέος χρήστης θα υποθέσει ότι η OpenGL είναι μια βιβλιοθήκη σχεδίασης γραφικών. Ωστόσο, με τον όρο OpenGL δεν αναφερόμαστε σε μια συγκεκριμένη βιβλιοθήκη αλλά σε ένα **πρότυπο υλοποίησης** βιβλιοθηκών σχεδίασης γραφικών. Εμπεριέχει δηλαδή το σύνολο των συναρτήσεων που πρέπει να υλοποιεί μία βιβλιοθήκη γραφικών προκειμένου να είναι συμβατή με αυτό. Το πρότυπο αυτό λοιπόν καθορίζει μια **προγραμματιστική διεπιφάνεια** (application programming interface ή API).

Είναι προφανές ότι δεν υπάρχει περιορισμός ως προς τη γλώσσα προγραμματισμού στην οποία θα υλοποιηθεί το πρότυπο της OpenGL. Ενδεικτικά αναφέρουμε την ύπαρξη βιβλιοθηκών σε γλώσσες προγραμματισμού Fortran και C. Τα παραδείγματα των Εργαστηριακών Μαθημάτων θα υλοποιηθούν σε γλώσσα προγραμματισμού C.

Εφόσον με το όρο OpenGL δεν αναφερόμαστε σε μια συγκεκριμένη βιβλιοθήκη αλλά σε ένα πρότυπο που ορίζει τη λειτουργικότητα μιας βιβλιοθήκης σχεδίασης, μπορούμε να ακολουθήσουμε τις ίδιες συμβάσεις σε όλες τις υλοποιήσεις του προτύπου (τις ίδιες εντολές). Αυτό σημαίνει ότι, εάν βασιστούμε στο πρότυπο της OpenGL, ο κώδικας που συντάσσουμε είναι **ανεξάρτητος πλατφόρμας (platform independent)** και μπορεί να εκτελεστεί σε ευρεία γκάμα περιβαλλόντων προγραμματισμού χωρίς ριζική τροποποίηση της δομής του.

Οι βιβλιοθήκες των περισσότερων νέων μεταγλωττιστών εμπεριέχουν (ή υπάρχει η δυνατότητα να ενσωματωθεί σε αυτούς) μια υλοποίηση της OpenGL.

### E.2 Βιβλιοθήκες

Στην ενότητα αυτή περιγράφουμε τις κατηγορίες βιβλιοθηκών που συναντά κανείς σε υλοποιήσεις της OpenGL, καθώς και συμβάσεις σε ό,τι αφορά το συμβολισμό εντολών και σταθερών.

#### α) Βασική βιβλιοθήκη (OpenGL core library):

Η βασική βιβλιοθήκη της OpenGL περιέχει τις κύριες εντολές σχεδίασης. Όλες 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 εισόδου-εξόδου. 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*.

## **E.3 Ένα τυπικό πρόγραμμα**

Στην παράγραφο αυτή παρουσιάζουμε και περιγράφουμε συνοπτικά τη δομή ενός τυπικού προγράμματος που βασίζεται στο πρότυπο της OpenGL.

```
#include <glut.h>

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;
}

```

Ας ξεκινήσουμε με την περιγραφή του κώδικα που περιέχεται στη συνάρτηση main.

- Αρχικά προσθέτουμε την κεφαλίδα του GLUT **glut.h**. Γενικά, για την ανάπτυξη ενός προγράμματος που χρησιμοποιεί ρουτίνες της OpenGL, απαιτείται η προσθήκη των κεφαλίδων **gl.h** και **glu.h**, ενώ για χρήση του GLUT, απαιτείται η προσθήκη της **glut.h**. Ωστόσο, με τη δήλωση της κεφαλίδας **glut.h** δηλώνονται αυτομάτως και οι δύο προηγούμενες.
- Η συνάρτηση **glutInit** ενεργοποιεί τη βιβλιοθήκη GLUT και μέσω αυτής μπορούμε να περάσουμε παραμέτρους στην εφαρμογή μας από τη γραμμή εντολών του DOS.
- Η ρουτίνα **glutInitWindowPosition** καθορίζει τη θέση στην οθόνη, στην οποία θα εμφανιστεί το παράθυρο της εφαρμογής (συντεταγμένη της άνω αριστερής κορυφής).
- Η εντολή **glutInitWindowSize** καθορίζει το πλάτος και ύψος του παραθύρου της εφαρμογής σε pixels.
- Η εντολή **glutInitDisplayMode** καθορίζει παραμέτρους σχετικές με τους ενταμιευτές και το χρωματικό μοντέλο που χρησιμοποιούνται κατά τη σχεδίαση. (Η τεχνική της διπλής ενταμίευσης χρησιμοποιείται κατά κόρον σε εφαρμογές κινουμένων γραφικών). Στο παράδειγμά μας χρησιμοποιούμε το χρωματικό μοντέλο RGB (**GLUT\_RGB**) και την εφαρμογή απλής ενταμίευσης (**GLUT\_SINGLE**).
- Η εντολή **glutCreateWindow** εμφανίζει το παράθυρο της εφαρμογής στην οθόνη και του αποδίδει έναν τίτλο.
- Με την εντολή **glMatrixMode(GLUint mode)** επιλέγουμε το μητρώο το οποίο επιθυμούμε να τροποποιήσουμε. (περισσότερα για τα μητρώα στις ενότητες «Μετασχηματισμοί συντεταγμένων» και «Προβολές») Στο παράδειγμα, δίνοντας ως όρισμα τη σταθερά **GL\_PROJECTION** επιλέγουμε το μητρώο προβολής, το οποίο καθορίζει τον τρόπο με τον οποίο προβάλλεται η σκηνή στο επίπεδο του θεατή. Σε συνδυασμό με την εντολή **gluOrtho2D(xMin,xMax,yMin,yMax)** διευκρινίζουμε ότι θα απεικονιστεί η **παράλληλη προβολή** της σκηνής στο επίπεδο XY (το οποίο, στο συγκεκριμένο παράδειγμα, ταυτίζεται με το παράθυρο της εφαρμογής). Το περιεχόμενο του παραθύρου περιλαμβάνει όλα τα στοιχεία του σκηνικού που εκτείνονται μεταξύ των συντεταγμένων X=[-50,50] και Y=[-50,50]. Ο θετικός άξονας X έχει φορά προς τα δεξιά και το θετικό τμήμα του άξονα Y έχει

φορά προς τα πάνω.

- Η εντολή ***glutDisplayFunc(void func())*** εντάσσεται σε μια ειδική κατηγορία συναρτήσεων του GLUT, οι οποίες αποκαλούνται **συναρτήσεις κλήσης** (callback functions). Η συγκεκριμένη συνάρτηση δέχεται ως όρισμα μια συνάρτηση στην οποία εμπεριέχεται ο κώδικας σχεδίασης γραφικών. Η συνάρτηση αυτή εκτελείται κάθε φορά που η εφαρμογή διαπιστώσει ότι απαιτείται επανασχεδιασμός της σκηνής. Η συνάρτηση δεν επιστρέφει τιμή και δεν έχει ορίσματα. Στο συγκεκριμένο παράδειγμα δίνουμε ως όρισμα τη συνάρτηση ***display()***.
- Η εντολή ***glutMainLoop()*** ενεργοποιεί τον **κύκλο διαχείρισης γεγονότων (event processing loop)**. Στον κύκλο αυτό, η εφαρμογή αναμένει επ' άπειρον και ανταποκρίνεται σε γεγονότα, όπως λ.χ. στο πάτημα ενός κουμπιού, στην αλλαγή του σκηνικού ή στην κίνηση του ποντικιού. Από το πρόθεμά της, παρατηρούμε ότι η συγκεκριμένη εντολή εμπεριέχεται στη βιβλιοθήκη GLUT, εφόσον το πρότυπο της OpenGL, ως πρότυπο ανεξάρτητο πλατφόρμας, δεν ορίζει διαδικασίες εισόδου-εξόδου.

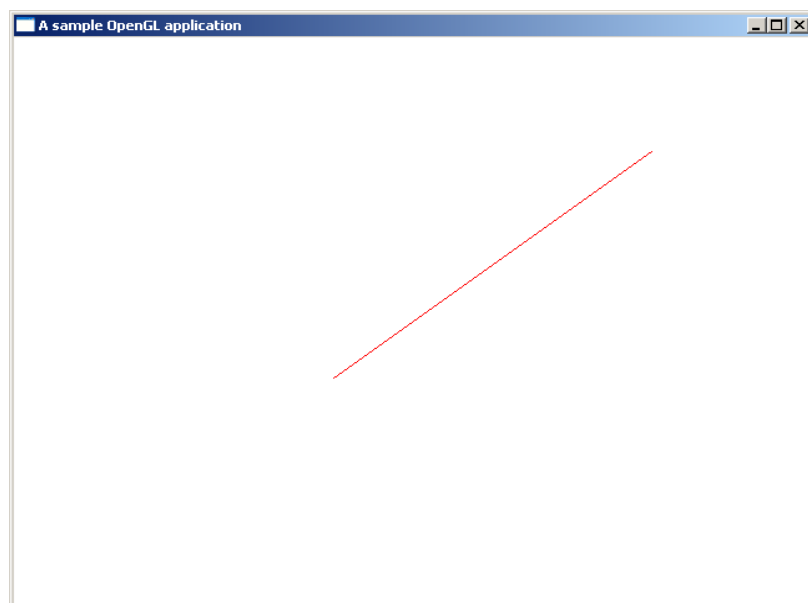
Ας αναλύσουμε τώρα το περιεχόμενο της συνάρτησης display.

- 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 σχεδίασης. Πρόκειται για μία ακόμη μεταβλητή κατάσταση 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 σχεδίασης το κόκκινο.
- Η εντολή ***glBegin(GLenum MODE)*** δηλώνει την έναρξη ορισμού ενός ή περισσότερων

γεωμετρικών σχημάτων. Αναλόγως του ορίσματος, μπορεί να προσδιοριστεί μια ποικιλία σχημάτων. Στο παράδειγμά μας, ορίζουμε ευθύγραμμα τμήματα. Η εντολή **glBegin** εκτελείται πάντα σε συνδυασμό με την εντολή **glEnd()** και η δεύτερη ορίζει τη λήξη της επιλεγόμενης ρύθμισης σχεδίασης.

- Η εντολή **glVertex2i** ορίζει σημεία στο διδιάστατο χώρο. Εφόσον έχει προεπιλεγεί η κατάσταση σχεδίασης ευθυγράμμων τμημάτων, τα σημεία ορίζουν ανά ζεύγη τα ευθύγραμμο τμήματα. Το παραπάνω παράδειγμα σχεδιάζει ένα ευθύγραμμο τμήμα κόκκινου χρώματος με συνταταγμένες αρχής και τέλους (20,20) (40,40).
- Η εντολή **glFlush()** εξαναγκάζει την εκτέλεση των εντολών που εκκρεμούν.

Η εκτέλεση του κώδικα εμφανίζει το παράθυρο του παρακάτω σχήματος.



Σχ. 1 Παράθυρο του Παραδείγματος

