

Εισαγωγή στην OpenGL

Περιεχόμενα εισαγωγικής ενότητας:

- Γενικά χαρακτηριστικά της OpenGL
- Βιβλιοθήκες της OpenGL
- Ένα τυπικό πρόγραμμα

Τι είναι η OpenGL;

- Η OpenGL δεν είναι μια συγκεκριμένη βιβλιοθήκη γραφικών.
- Είναι ένα πρότυπο που καθορίζει τις λειτουργίες που πρέπει να υποστηρίζει μία βιβλιοθήκη γραφικών ούτως ώστε να είναι συμβατή με αυτήν.
- Η OpenGL ορίζει μια προγραμματιστική διεπιφάνεια (Application Programming Interface) σχεδίασης γραφικών.

Χαρακτηριστικά της OpenGL

- Ο προγραμματιστής προγραμματίζει χρησιμοποιώντας μια υλοποίηση της OpenGL.
- Μια βιβλιοθήκη που υλοποιεί το πρότυπο της OpenGL μπορεί να συνταχθεί σε οποιαδήποτε γλώσσα προγραμματισμού (η OpenGL είναι πρότυπο ανεξάρτητο πλατφόρμας).
- Οι περισσότεροι μεταγλωττιστές εμπεριέχουν ή μπορεί να ενσωματωθεί σε αυτούς μία βιβλιοθήκη της OpenGL.
- Εμείς θα χρησιμοποιήσουμε την υλοποίηση της OpenGL στη γλώσσα προγραμματισμού C++.

Κατηγορίες βιβλιοθηκών της OpenGL

•GL Core Library: Περιέχει βασικές εντολές σχεδίασης (σχεδίαση βασικών γεωμετρικών σχημάτων, ορισμός χρωμάτων κλπ.) Όλες οι ρουτίνες της βιβλιοθήκης ξεκινούν με το πρόθεμα **gl**

OpenGL Utility Library (GLU): Περιέχει πιο εξειδικευμένες λειτουργίες (Σχεδίαση σύνθετων επιφανειών, ορισμός προβολών κλπ) Όλες οι ρουτίνες της βιβλιοθήκης ξεκινούν με το πρόθεμα **glu**

OpenGL Utility Toolkit (GLUT): Ρουτίνες εισόδου-εξόδου (σχηματισμός παραθύρων, διαχείριση γεγονότων κλπ.). Κατ' εξαίρεση, οι εντολές της GLUT εξαρτώνται από την αρχιτεκτονική του συστήματος. Όλες οι εντολές της βιβλιοθήκης ξεκινούν με το πρόθεμα **glut**.

Ένα τυπικό παράδειγμα

```
#include <GL\glut.h>
```

```
void display()
```

```
{  
    glClearColor(1, 1, 1, 0);  
    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;}
```

Κεφαλίδες της OpenGL

`#include <GL\gl.h>` = OpenGL Core Library

`#include <GL\glu.h>` = OpenGL Utility

`#include <GL\glut.h>` = OpenGL Core Library +
OpenGL Utility +
OpenGL Utility Toolkit

Ανάλυση παραδείγματος (main) (1)

glutInit(): Ενεργοποιεί τη χρήση των εντολών της βιβλιοθήκης GLUT.



```
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;}  
}
```

Ανάλυση παράδειγματος (main) (2)

glutInitWindowPosition:

Δηλώνει τη θέση στην οθόνη όπου θα εμφανιστεί το παράθυρο της εφαρμογής



```
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;}
```


Ανάλυση παράδειγματος (main) (3)

glutInitWindowSize:

Ορίζει τις διαστάσεις του παραθύρου της εφαρμογής σε pixels



```
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;}
```

Ανάλυση παράδειγματος (main) (4)

glutInitDisplayMode:

Καθορίζει ρυθμίσεις απεικόνισης (μοντέλο ενταμίευσης, χρωματικό μοντέλο κ.λ.π.)



```
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;}
```

Ανάλυση παράδειγματος (main) (5)

```
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;}  
}
```

glutCreateWindow:

Εμφανίζει το παράθυρο της εφαρμογής



Ανάλυση παράδειγματος (main) (6)

```
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;}
```

glMatrixMode: Επιλέγει το μητρώο που επιθυμούμε να τροποποιήσουμε (προβολής ή μετασχηματισμού μοντέλου)



Ανάλυση παραδείγματος (main) (7)

```
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;}
```

gluOrtho2D: Δήλωση
παράλληλης προβολής



Ανάλυση παράδειγματος (main) (8)

```
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;}
```

glutDisplayFunc:

Δηλώνει τη συνάρτηση που θα εκτελείται κάθε φορά που απαιτείται σχεδιασμός της σκηνής



glutDisplayFunc(display);

Ανάλυση παράδειγματος (main) (9)

```
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;}

```

glutMainLoop:

Ενεργοποιεί τον κύκλο
ακρόασης γεγονότων



glutMainLoop();

Ανάλυση παραδείγματος (display) (1)

glClearColor:

Δήλωση χρώματος
καθαρισμού της
οθόνης



```
void display()
```

```
{
```

```
glClearColor(1,1,1,0);
```

```
glClear(GL_COLOR_BUFFER_BIT);
```

```
glColor3f(1,0,0);
```

```
glBegin(GL_LINES);
```

```
glVertex2i(20,20);
```

```
glVertex2i(40,40);
```

```
glEnd();
```

```
glFlush();
```

```
}
```


Ανάλυση παραδείγματος (display) (2)

glClear: Καθαρισμός
οθόνης (καθαρισμός ενός
από τους ενταμιευτές του
συστήματος γραφικών)



```
void display()
{
    glClearColor(1,1,1,0);
    glClear(GL_COLOR_BUFFER_BIT);

    glColor3f(1,0,0);

    glBegin(GL_LINES);
    glVertex2i(20,20);
    glVertex2i(40,40);
    glEnd();

    glFlush();
}
```

Ανάλυση παραδείγματος (display) (3)

```
void display()
{
    glClearColor(1,1,1,0);
    glClear(GL_COLOR_BUFFER_BIT);

    glColor3f(1,0,0);

    glBegin(GL_LINES);
    glVertex2i(20,20);
    glVertex2i(40,40);
    glEnd();

    glFlush();
}
```

glColor*: Επιλογή
χρώματος σχεδίασης



glColor3f(1,0,0);

Ανάλυση παραδείγματος (display) (4)

```
void display()
{
    glClearColor(1,1,1,0);
    glClear(GL_COLOR_BUFFER_BIT);

    glColor3f(1,0,0);

    glBegin(GL_LINES);
    glVertex2i(20,20);
    glVertex2i(40,40);
    glEnd();

    glFlush();
}
```

glBegin/glEnd: Μεταξύ αυτών των εντολών δηλώνονται συντεταγμένες κορυφών γεωμετρικών σχημάτων. Το είδος των σχημάτων καθορίζεται από το όρισμα της `glBegin`.

Ανάλυση παραδείγματος (display) (5)

```
void display()
{
    glClearColor(1,1,1,0);
    glClear(GL_COLOR_BUFFER_BIT);

    glColor3f(1,0,0);

    glBegin(GL_LINES);
    glVertex2i(20,20);
    glVertex2i(40,40);
    glEnd();

    glFlush();
}
```

`glVertex*`: Δήλωση
συντεταγμένων
μεμονωμένων κορυφών.



Ανάλυση παραδείγματος (display) (6)

```
void display()
{
    glClearColor(1,1,1,0);
    glClear(GL_COLOR_BUFFER_BIT);

    glColor3f(1,0,0);

    glBegin(GL_LINES);
    glVertex2i(20,20);
    glVertex2i(40,40);
    glEnd();
}
```

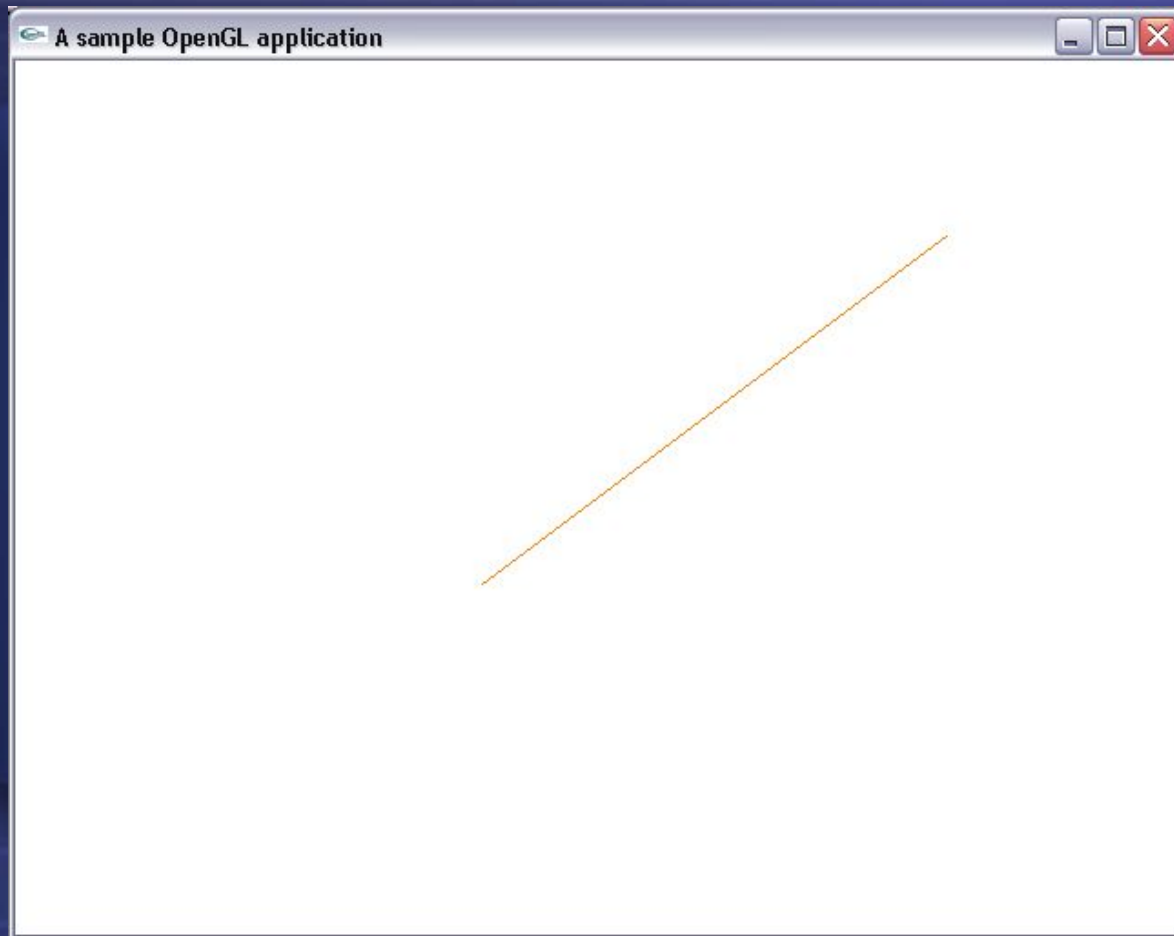
glFlush: Προωθεί την εκτέλεση εντολών που εκκρεμούν.



```
glFlush();
```

```
}
```

Αποτέλεσμα παραδείγματος



Τέλος ενότητας!

