

الجلسة الخامسة: تعليمات لوحة المفاتيح والفأرة

الهدف من الجلسة

- التعرف بأوامر لوحة المفاتيح والفأرة.
- حل تمارين

تعليمات لوحة المفاتيح

يوجد نوعين من تعليمات التعامل مع لوحة المفاتيح:

تعليمة التعامل مع الضغط على مفاتيح لوحة المفاتيح العامة.

تستخدم للتعامل مع مفاتيح لوحة المفاتيح العامة مثل الأحرف والأرقام، ويتم ذلك عن طريق:

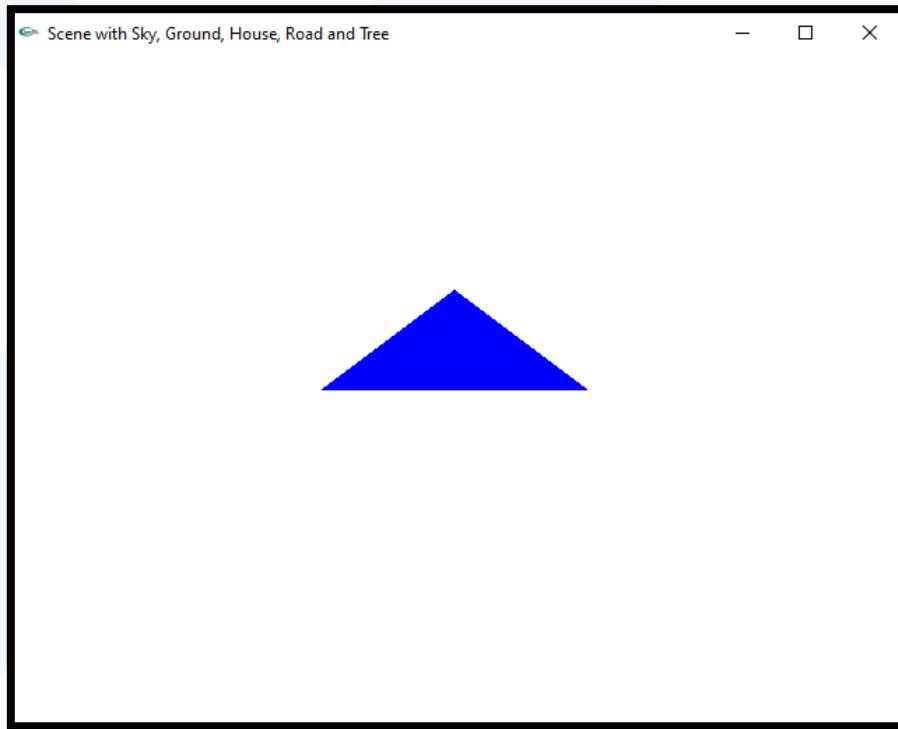
1. تعريف تابع `keyboard(unsigned char key, int x, int y)` يمكنه الوصول لمفاتيح لوحة التحكم.



2. كتابة تعليمة `glutPostRedisplay()` في نهاية تابع `keyboard`.
3. استدعاء هذا التابع عن طريق تعليمة `glutKeyboardFunc(keyboard)` في الـ `main`.

مثال:

ليكن لدينا الشكل التالي:



ونريد تحريكه نحو اليمين عند الضغط على الحرف 'd' ونحو اليسار عند الضغط على الحرف 'a' ونحو الأعلى عند الضغط على 'w' ونحو الأسفل عند الضغط على 's'.

سننتبع الخطوات التالي:

(1) تعريف متحولات عامة global يتم تغيير قيمتها ضمن تابع keyboard حسب الحرف المضغوط:

للتحرك نحو اليمين واليسار (moveX) والتحرك نحو الأعلى والأسفل (moveY).

(2) تعريف تابع keyboard واختبار حالة المفتاح المضغوط كالتالي:

- عند الضغط على الحرف 'd' سيتأثر المتغير الخاص بالتحريك على محور x بالزيادة.
- عند الضغط على الحرف 'a' سيتأثر المتغير الخاص بالتحريك على محور x بالنقصان.
- عند الضغط على الحرف 'w' سيتأثر المتغير الخاص بالتحريك على محور y بالزيادة.
- عند الضغط على الحرف 's' سيتأثر المتغير الخاص بالتحريك على محور y بالنقصان.

(3) استخدام تابع التحريك مع المتحولات العامة قبل الشكل المراد تحريكه.

(4) تفعيل التابع keyboard في الـ main عن طريق التعليمة:

glutKeyboardFunc(keyboard)

الحل البرمجي:

```
#include <GL/glut.h>
int moveX=0,moveY=0;
void draw2DTriangle(){
glBegin(GL_TRIANGLES);
glColor3f(0,0,1);
glVertex2d(-3,0);
glVertex2d(0,3);
glVertex2d(3,0);
glEnd();
}

void keyboard(unsigned char key,int x,int y)
{
if(key=='d')
    moveX++;
if(key=='a')
    moveX--;
if(key=='w')
    moveY++;
```

```
if(key=='s')
    moveY--;
glutPostRedisplay();
}
void myDisplay(void)
{
    glClearColor(1,1,1,1);
    glClear(GL_COLOR_BUFFER_BIT);
    glTranslatef(moveX,moveY,0);
    draw2DTriangle();
    glutSwapBuffers();
}
int main(int argc, char **argv)
{
    glutInit(&argc, argv);
    glutInitDisplayMode(GLUT_DOUBLE | GLUT_RGB | GLUT_DEPTH);
    glutInitWindowSize(640, 480);
    glutInitWindowPosition(10, 10);
    glutCreateWindow("Lab5");
    glOrtho(-10, 10, -10, 10, -10, 10);
    glutKeyboardFunc(keyboard);
    glutDisplayFunc(myDisplay);
    glutMainLoop();
    return 0;
}
```

إعادة رسم النافذة
الحالية

تعليلة التعامل مع الضغط على مفاتيح ذات أغراض خاصة

تستخدم للتعامل مع مفاتيح ذات أغراض محددة مثل الأسهم و زر Esc و pgup و pgdn وغيرها ويتم ذلك عن طريق:

1. تعريف تابع `specialKeys(int key, int x, int y)` يمكنه الوصول لمفاتيح لوحة التحكم.



2. كتابة تعليلة `glutPostRedisplay()` في نهاية تابع `specialKeys`.

3. استدعاء هذا التابع عن طريق تعليلة `glutSpecialFunc(specialKeys)` في الـ `main`.

المفاتيح الخاصة:

**GLUT_KEY_F(1...12), GLUT_KEY_LEFT, GLUT_KEY_UP,
GLUT_KEY_RIGHT, GLUT_KEY_DOWN, GLUT_KEY_PAGE_UP,
GLUT_KEY_PAGE_DOWN, GLUT_KEY_HOME, GLUT_KEY_END,
GLUT_KEY_INSERT**

مثال:

إذا أردنا تحريك المثلث السابق لكن باستخدام الأسهم، فنقوم بنفس الخطوات السابقة مع تغيير التوابيع، كما يلي:

```
#include <GL\glut.h>
int moveX=0,moveY=0;
void draw2DTriangle(){
glBegin(GL_TRIANGLES);
glColor3f(0,0,1);
glVertex2d(-3,0);
glVertex2d(0,3);
```

```
glVertex2d(3,0);
glEnd();
}

void sepcialKey(int key,int x,int y){
if(key==GLUT_KEY_RIGHT)
    moveX++;
if(key==GLUT_KEY_LEFT)
    moveX--;
if(key==GLUT_KEY_UP)
    moveY++;
if(key==GLUT_KEY_DOWN)
    moveY--;
glutPostRedisplay();
}

void myDisplay(void)
{
glClearColor(1,1,1,1);
glClear(GL_COLOR_BUFFER_BIT);
glTranslatef(moveX,moveY,0);
draw2DTriangle();

glutSwapBuffers();
}

int main(int argc, char **argv)
{
glutInit(&argc, argv);
glutInitDisplayMode(GLUT_DOUBLE | GLUT_RGB | GLUT_DEPTH);
glutInitWindowSize(640, 480);
```

```

glutInitWindowPosition(10, 10);
glutCreateWindow("Lab5");
glOrtho(-10, 10, -10, 10, -10, 10);
glutSpecialFunc(sepcialKey);
glutDisplayFunc(myDisplay);
glutMainLoop();
return 0;
}

```

تدريب 1:

- اجعل الشكل السابق يدور مع عقارب الساعة عند الضغط على السهم اليميني وعقارب الساعة عند الضغط على السهم اليساري.
- اجعل الشكل يزداد بسرعة دورانه عند الضغط على المفتاح 'f'.

تعليمات الفأرة

للتعامل مع الفأرة سنتعامل مع ثلاث توابع أساسية:

تعليمة التعامل مع الضغط على أحد أزرار الفأرة.

يتم ذلك بالشكل التالي:

1. تعريف تابع (`mouseClick(int button, int state, int x, int y)` لتحديد الزر المضغوط من الفأرة.



2. كتابة تعليمة (`glutPostRedisplay()` في نهاية تابع `mouseClick`.

3. استدعاء هذا التابع عن طريق تعليمة (`glutMouseFunc(mouseClick)` في الـ `main`.

حالات الـ button:

الضغط على الزر اليميني: GLUT_RIGHT_BUTTON

الضغط على الزر اليساري: GLUT_LEFT_BUTTON

الضغط على الزر الأوسط: GLUT_MIDDLE_BUTTON

حالات الـ state:

حالة الضغط: GLUT_DOWN

حالة رفع الضغط: GLUT_UP

مثال:

يُراد تحريك الشكل السابق نحو اليمين عند الضغط على الزر الأيمن للفأرة وتحريكه نحو اليسار عند الضغط على الزر الأيسر للفأرة.

الحل:

```
#include <GL\glut.h>
int moveX=0,moveY=0;
void draw2DTriangle(){
glBegin(GL_TRIANGLES);
glColor3f(0,0,1);
glVertex2d(-3,0);
glVertex2d(0,3);
glVertex2d(3,0);
glEnd();
}
```

```
void mouseClick(int button,int state,int x,int y) {
if(button==GLUT_RIGHT_BUTTON && state==GLUT_DOWN)
```

```
        moveX++;  
if(button==GLUT_LEFT_BUTTON && state==GLUT_DOWN)  
    moveX--;  
glutPostRedisplay();  
}  
  
void myDisplay(void)  
{  
    glClearColor(1,1,1,1);  
    glClear(GL_COLOR_BUFFER_BIT);  
    glTranslatef(moveX,0,0);  
    draw2DTriangle();  
  
    glutSwapBuffers();  
}  
  
int main(int argc, char **argv)  
{  
    glutInit(&argc, argv);  
    glutInitDisplayMode(GLUT_DOUBLE | GLUT_RGB | GLUT_DEPTH);  
    glutInitWindowSize(640, 480);  
    glutInitWindowPosition(10, 10);  
    glutCreateWindow("Lab5");  
    glOrtho(-10, 10, -10, 10, -10, 10);  
    glutMouseFunc(mouseClick);  
    glutDisplayFunc(myDisplay);  
    glutMainLoop();  
    return 0;}
```

تعليلة التعامل مع تحريك الفأرة مع الضغط.

يتم ذلك بالشكل التالي:

1. تعريف تابع `mouseMove(int x, int y)` للحصول على إحداثيات الفأرة.



2. كتابة تعليلة `glutPostRedisplay()` في نهاية تابع `mouseMove`.

3. استدعاء هذا التابع عن طريق تعليلة `glutMotionFunc(mouseMove)` في الـ `main`.

مثال:

يُراد تغيير لون الشكل عند الضغط داخل المثلث ليصبح أحمر وعند الضغط خارج المثلث يصبح لونه أصفر.

خطوات الحل:

- التحويل من قيم الشاشة (الطول والعرض) إلى إحداثيات الـ `ortho`.
- اختبار الضغط ضمن مجال الـ `x` للشكل ومجال الـ `y`.
- تغيير القيم الموجودة في تابع `glColor3f` المكتوب داخل تابع رسم المثلث.

الحل البرمجي:

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

```
#include<stdio.h>
```

```
int r=0,g=0,b=1;
```

```
int winW=640,winH=480;
```

```
void mouseMove(int x,int y){
```

```
float rX=(float)x/winW *20-10;
```

```
float rY=10-20*(float)y/winH;
```

```
if((rX>=-3 &&rX<=3)&&(rY>=0 && rY<=3))
```

```
{
```

```
    r=1;  
    g=0;  
    b=0;  
}  
else{  
    r=1;  
    g=1;  
    b=0;  
}
```

```
glutPostRedisplay();  
}
```

```
void draw2DTriangle(){  
    glBegin(GL_TRIANGLES);  
    glColor3f(r,g,b);  
    glVertex2d(-3,0);  
    glVertex2d(0,3);  
    glVertex2d(3,0);  
    glEnd();  
}
```

```
void myDisplay(void)  
{  
    glClearColor(1,1,1,1);  
    glClear(GL_COLOR_BUFFER_BIT);  
    draw2DTriangle();  
}
```

```
glutSwapBuffers();  
}  
  
int main(int argc, char **argv)  
{  
    glutInit(&argc, argv);  
    glutInitDisplayMode(GLUT_DOUBLE | GLUT_RGB | GLUT_DEPTH);  
    glutInitWindowSize(winW, winH);  
    glutInitWindowPosition(10, 10);  
    glutCreateWindow("Lab5");  
    glOrtho(-10, 10, -10, 10, -10, 10);  
    glutMotionFunc(mouseMove);  
    glutDisplayFunc(myDisplay);  
  
    glutMainLoop();  
    return 0;  
}
```

يوجد تابع ثالث للمرور على الشكل دون الضغط عليه.

تدريب 2:

ارسم مثلث في مكان الضغط على الشاشة.

الحل:

https://drive.google.com/drive/folders/1FkDfROA1JatyHNbLtd_dPXVHKH1MZ4aQ?usp=sharing

انتهت الجلسة