

键盘、鼠标交互

(事件驱动机制)

回调函数 callback function

- 用户定义用来处理特定事件的函数
- `glutMainLoop()` ;
- Window: `resize.....`
- Mouse: click one or more buttons
- Motion: move mouse
- Keyboard: press or release a key
- Idle: nonevent

display callback

- The display callback is executed whenever GLUT determines that the window should be refreshed, for example
 - When the window is first opened
 - When the window is reshaped
 - When a window is exposed
 - When the user program decides it wants to change the display
- In `main.c`
 - `glutDisplayFunc(mydisplay)` identifies the function to be executed
 - Every GLUT program must have a display callback

实验内容

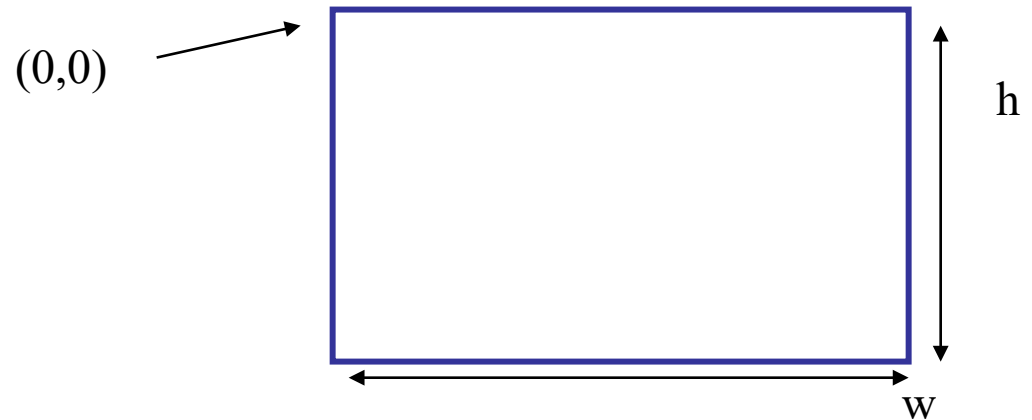
- 编写交互程序，要求：
- 1. 显示基本线段，三角形，四边形，多边形等图形。
- 2. 用户可以通过按键来控制绘制的图形类型和颜色。如：t 绘制三角形，r 图形改为红色)
- 3. 使用键盘控制图形的位置。
- 4. 用户可以通过点击鼠标左右键来调整图形的类型和颜色。

鼠标

- `glutMouseFunc(mymouse)`
- `void mymouse(GLint button, GLint state, GLint x, GLint y)`
- 其中 `button` 的值可能是
 - `GLUT_LEFT_BUTTON`,
 - `GLUT_MIDDLE_BUTTON`,
 - `GLUT_RIGHT_BUTTON`
- `state` 表示相应按钮的状态：
 - `GL_UP`, `GL_DOWN`
- `x, y` 表示在窗口中的位置

位置

- 屏幕坐标以像素为基本单位，原点在左上角。
- OpenGL 坐标系原点在左下角。
- Must invert y coordinate returned by callback by height of window
 - $y = h - y;$



鼠标函数

- `glutMotionFunc((void*)func(int x, int y));`
- `glutPassiveMotionFunc((void*)func(int x, int y));`
- Active motion 是指鼠标移动并且有一个鼠标键被按下。
- Passive motion 是指当鼠标移动时，并没有鼠标键按下

参考代码 - 鼠标

- 主函数增加注册鼠标事件函数
- `glutMouseFunc(&OnMyMouse);`

```
void OnMyMouse(int button,int state,int x,int y)
{
    if(button==GLUT_LEFT_BUTTON&&state==GLUT_LEFT_BUTTON_DOWN)
    {
        /* 自定义 */
    }

    if(button==GLUT_RIGHT_BUTTON&&state==GLUT_RIGHT_BUTTON_DOWN)
    {
        /* 自定义 */
    }
}
```


键盘

- `glutKeyboardFunc(mykey)`
- `void mykey(GLubyte key, GLint x, GLint y)`
- `key` 返回键盘上被按下键的 ASCII 码和鼠标位置
- 注意在 GLUT 中并不把释放键做为一个事件
- `x` , `y`, 鼠标坐标

参考代码 - 键盘

- `void mykey(GLubyte key, GLint x, GLint y)`
- `{`
- `if(key == 'Q' | key == 'q') exit(0);`
- `}`

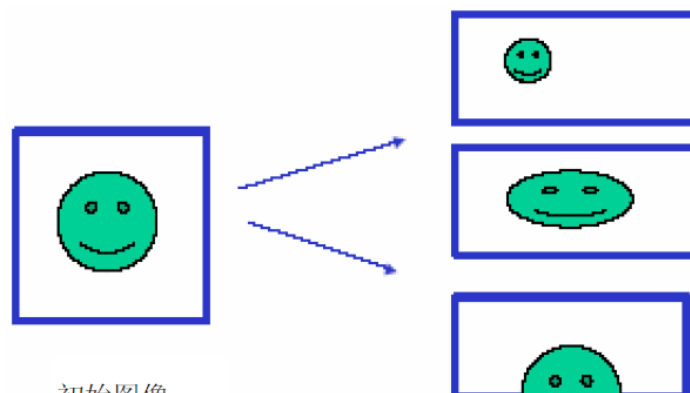
```
void OnMyKeyboards(unsigned char key,int x,int y )
{
    switch (key)
    {
        case 'w':/* 处理 */
            break;
    }
}
```

键盘相关函数

- GLUT 在 glut.h 中定义了特殊按键：
 - 功能键 1: GLUT_KEY_F1
 - 向上方向键：GLUT_KEY_UP
 - if(key == GLUT_KEY_F1)
- glutGetModifiers(...) 探测是否按下了
 - GLUT_ACTIVE_SHIFT,
 - GLUT_ACTIVE_CTRL,
 - GLUT_ACTIVE_ALT

窗口形状的改变：

- 用户通过拖动改变窗口的形状和尺寸时，显示内容重新绘制？
- 显示原来内容的一部分
- 通过强迫适应新窗口来显示所有内容，这时，显示长宽比率可能会改变。



初始图像

改变后的图像

窗口改变的回调函数

- `glutReshapeFunc(myreshape)`
- `void myreshape(GLint w, GLint h)`

保持长宽比率不变

```
void myreshape(GLint w, GLint h) {  
    glViewport(0, 0, w, h);  
    glMatrixMode(GL_PROJECTION);  
    glLoadIdentity();  
    if(w<=h)  
        gluOrtho2D(-2.0, 2.0, -2.0*(GLfloat)h/(GLfloat)w,  
                    2.0*(GLfloat)h/(GLfloat)w);  
    else gluOrtho2D(-2.0*(GLfloat)w/(GLfloat)h,  
                    2.0*(GLfloat)w/(GLfloat)h, -2.0, 2.0);  
    glMatrixMode(GL_MODELVIEW);  
}
```