```python
from OpenGL.GL import *
from OpenGL.GLUT import *
from OpenGL.GLU import *
import time

# Global variables for triangle position and movement direction
x_pos = -1.0
y_pos = 0.0
speed = 0.001
direction_x = 1  # 1 for right, -1 for left
direction_y = 0  # 1 for up, -1 for down, 0 for no movement

# Display callback function
def display():
    global x_pos, y_pos
    # Clear the background
    glClear(GL_COLOR_BUFFER_BIT)

    # Draw the triangle
    glLoadIdentity()
    glTranslatef(x_pos, y_pos, 0)
    glColor3f(1, 0, 0)
    glBegin(GL_TRIANGLES)
    glVertex3f(-0.1, -0.1, 0)
    glVertex3f(0.1, -0.1, 0)
    glVertex3f(0, 0.1, 0)
    glEnd()

    # Swap buffers
    glutSwapBuffers()

# Idle callback function
def idle():
    global x_pos, y_pos, direction_x, direction_y, speed

    # Update the position of the triangle
    x_pos += direction_x * speed
    y_pos += direction_y * speed

    # Check for wrapping behavior
    if x_pos > 1.0:
        x_pos = -1.0
    elif x_pos < -1.0:
        x_pos = 1.0
    if y_pos > 1.0:
        y_pos = -1.0
    elif y_pos < -1.0:
        y_pos = 1.0

    # Request redisplay
    glutPostRedisplay()
```

```python
# Keyboard callback function for arrow keys
def special_keys(key, x, y):
    global direction_x, direction_y

    if key == GLUT_KEY_RIGHT:
        direction_x = 1
        direction_y = 0
    elif key == GLUT_KEY_LEFT:
        direction_x = -1
        direction_y = 0
    elif key == GLUT_KEY_UP:
        direction_x = 0
        direction_y = 1
    elif key == GLUT_KEY_DOWN:
        direction_x = 0
        direction_y = -1

# Main function
def main():
    glutInit()
    glutInitDisplayMode(GLUT_DOUBLE | GLUT_RGB)
    glutInitWindowSize(500, 500)
    glutCreateWindow("Triangle Animation")

    # Set the clear color
    glClearColor(0.0, 0.0, 0.0, 1.0)

    # Set the display callback
    glutDisplayFunc(display)
    # Set the idle callback
    glutIdleFunc(idle)
    # Set the special keys callback
    glutSpecialFunc(special_keys)

    # Start the GLUT event loop
    glutMainLoop()

# Run the program
if __name__ == "__main__":
    main()
```

Animated Triangle with Keyboard Control in OpenGL with Python

This code demonstrates creating an animated triangle that bounces around the screen using keyboard controls for direction changes.

Global Variables:

- `x_pos`, `y_pos`: Track the current x and y coordinates of the triangle's position.
- `speed`: Defines the movement speed of the triangle.

- `direction_x`, `direction_y`: Represent the movement direction (1 for right/up, -1 for left/down, 0 for no movement) on the x and y axes, respectively.

Display Callback Function (display())

- Updates the global `x_pos` and `y_pos` variables.
- `glClear(GL_COLOR_BUFFER_BIT)`: Clears the background color (black in this case).
- `glLoadIdentity()`: Resets the Modelview matrix to the identity matrix for proper positioning.
- `glTranslatef(x_pos, y_pos, 0)`: Applies a translation transformation to move the triangle based on its current coordinates.
- `glColor3f(1, 0, 0)`: Sets the drawing color to red for the triangle.
- Defines and draws the triangle using `glBegin(GL_TRIANGLES)` and `glVertex3f` for its vertices.
- `glutSwapBuffers()`: Swaps the front and back buffers for smooth animation.

Idle Callback Function (idle())

- This function is called continuously between frames when there's nothing else to render, creating an animation effect.
- Updates the triangle's position based on its current coordinates, `speed`, and movement direction (`direction_x` and `direction_y`).
- Implements edge bouncing behavior:
    - Checks if the triangle's position exceeds the window boundaries (-1.0 to 1.0) on the x and y axes.
    - If a boundary is crossed, the position is flipped to the opposite side of the window, creating a bouncing effect.
- `glutPostRedisplay()`: Requests a redisplay of the scene, triggering the `display()` function again.

Keyboard Callback Function (special_keys())

- Handles arrow key presses for changing the movement direction of the triangle.
- Based on the pressed key (GLUT_KEY_RIGHT, GLUT_KEY_LEFT, GLUT_KEY_UP, GLUT_KEY_DOWN), it updates the `direction_x` and `direction_y` variables to reflect the new movement direction.

Main Function (main())

- Performs standard GLUT initialization steps for window creation, display mode configuration, and callback assignment.
- Sets the clear color to black using `glClearColor`.
- Defines the callback functions for display, idle updates, and special key presses.
- Starts the main GLUT event loop (`glutMainLoop()`) to handle events and rendering continuously.

Running the Program

- The code is wrapped in an `if __name__ == "__main__":` block to ensure the `main()` function is only executed when this script is run directly.

Summary

This code demonstrates the combined use of display, idle, and keyboard callback functions to create an interactive animation in OpenGL with Python. You can adjust the `speed` variable to control the movement speed of the triangle.