Interaction in OpenGL

OpenGL GLUT Callback Functions
- GLUT's version of event/message handling
- Programmer specifies function to be called by OS in response to different events
- Specify the function by using glut***Func(ftn)
  - We've already seen glutDisplayFunc(disp_ftn)
    - disp_ftn called when client area needs to be repainted
    - Like Windows response to WM_PAINT messages
  - All GLUT callback functions work like MFC On***() event handler functions

Some Other GLUT Callbacks

- glutReshapeFunc(ftn(width,height))
  - Identifies function ftn() invoked when user changes size of window
    - height & width of new window returned to ftn()
- glutKeyboardFunc(ftn(key,x,y))
  - Identifies function ftn() invoked when user presses a keyboard key
    - Character code (key) and position of mouse cursor (x,y) returned to ftn()
- glutSpecialFunction(ftn(key,x,y))
  - For special keys such as function & arrow keys

Mouse Callbacks

- glutMouseFunc(ftn(button, state, x, y))
  - Identifies function ftn() called when mouse events occur
    - Button presses or releases
    - Position (x,y) of mouse cursor returned
    - Also the state (GLUT_UP or GLUT_DOWN)
    - Also which button
      - GLUT_LEFT_BUTTON, GLUT_RIGHT_BUTTON, or GLUT_MIDDLE_BUTTON

Mouse Motion

- Move event: when mouse moves with a button pressed –
  - glutMotionFunctionFunc(ftn(x,y))
    - ftn(x,y) called when there's a move event
    - Position (x,y) of mouse cursor returned
- Passive motion event: when mouse moves with no button pressed
  - glutPassiveMotionFunctionFunc(ftn(x,y))
    - ftn(x,y) called when there's a passive motion event
    - Position (x,y) of mouse cursor returned
GLUT Menus

- Can create popup menus and add menu items with:
  - glutCreateWindow (menu-ftn(ID))
    - Menu-ftn(ID) is callback function called when user selects an item from the menu
    - ID identifies which item was chosen
  - glutAddMenuEntry (name, ID_value)
    - Adds an entry with name displayed to current menu
    - ID_value returned to menu_ftn() callback
  - glutAttachMenu (button)
    - Attaches current menu to specified mouse button
    - When that button is pressed, menu pops up

Hierarchical Models

- In many applications the parts of a model depend on each other
- Often the parts are arranged in a hierarchy
  - Represent as a tree data structure
  - Transformations applied to parts in parent nodes are also applied to parts in child nodes
  - Simple example: a robot arm
    - Base, lower arm, and upper arm
      - Base rotates lower and upper arm also rotate
      - Lower arm rotates upper arm also rotates

Simple Robot Arm Hierarchical Model

Use of Matrix Stacks in OpenGL to Implement Hierarchies

- Matrix stacks store projection & model-view matrices
- Push and pop matrices with:
  - glPushMatrix();
  - glPopMatrix();
- Can use to position entire object while also preserving it for drawing other objects
- Use in conjunction with geometrical transformations
- Example: Robot program

OpenGL Hierarchical Models

- Set up a hierarchical representation of scene (a tree)
- Each object is specified in its own modeling coordinate system
- Traverse tree and apply transformations to bring objects into world coordinate system
- Traversal rule:
  - Every time we go to the left at a node with another unvisited right child, do a push
  - Every time we return to that node, do a pop
  - Do a pop at the end so number of pushes & pops are the same

GLUT Animation

- Simple method is to use an "idle" callback
  - Called whenever window’s event queue is empty
  - Could be used to update display with the next frame of the animation
  - Identify the idle function with:
    - glutIdleFunc(idle_ftn())
  - Simple Example:
    void idle_ftn()
    {
      glutPostRedisplay();
    }
    - Posts message to event queue that client area needs to be repainted
    - Causes display callback function to be invoked
    - Effectively displays next frame of animation
Double Buffering

- Use two display buffers
- Front buffer is displayed by display hardware
- Application draws into back buffer
- Swap buffers after new frame is drawn into back buffer
- Implies only one access to display hardware per frame
- Eliminates flicker
- In OpenGL, implement by replacing glFlush() with glutSwapBuffers() in display callback
- In initialization function, must use:
  glutInitDisplayMode(GLUT_DOUBLE | GLUT_RGB);
- See anim_square & cone.anim examples