CSE 3200 Micro-Computer Graphics Introducing GLUT
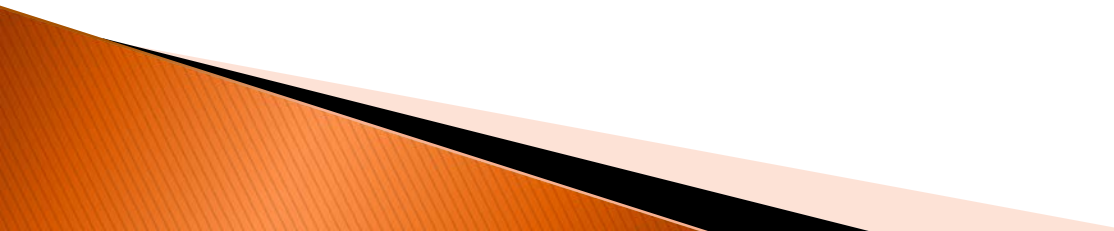
Presenter: Girendra Persaud

University of Guyana

Outline

- About GLUT
- Controls, Input, Windowing
- Window System Routines
- Input
- Controls
- Questions?
- Further Reading

What is GLUT?

- GLUT is the OpenGL Utility Toolkit, a window system independent toolkit for writing OpenGL programs.

- GLUT is designed for constructing small to medium sized OpenGL programs.

- The GLUT source code distribution is portable to nearly all OpenGL implementations and platforms.

- The current version is 3.7. Additional releases of the library are not anticipated.

- GLUT is not open source. Mark Kilgard maintains the copyright.

- The current version of the GLUT API is 3.

Control, Input & Windowing

- The toolkit supports:
  - Multiple windows for OpenGL rendering
  - Callback driven event processing
  - Sophisticated input devices
  - An 'idle' routine and timers
  - A simple, cascading pop-up menu facility
  - Utility routines to generate various solid and wire frame objects
  - Support for bitmap and stroke fonts
  - Miscellaneous window management functions

Window System Routines

▸ **glutInit**(int *argc, char **argv) initializes GLUT and processes any command line arguments. **glutInit()** should be called before any other GLUT routine.

▸ **glutInitDisplayMode**(unsigned int mode)
  ◦ specifies whether to use an *RGBA* or color-index color model.
  ◦ If you want a single- or double-buffered window.
  ◦ Specify depth, stencil, and/or accumulation buffer.
  ◦ For example,
    • **glutInitDisplayMode**(GLUT_DOUBLE | GLUT_RGB | GLUT_DEPTH).

Window System Routines

- **glutInitWindowPosition**(int x, int y) specifies the screen location for the upper-left corner of your window.

- **glutInitWindowSize**(int width, int size) specifies the size, in pixels, of your window.

- int **glutCreateWindow**(char *string) creates a window with an OpenGL context. It returns a unique identifier for the new window. Be warned: Until **glutMainLoop()** is called, the window is not yet displayed

Input

- **void glutKeyboardFunc(char key,int x, int y);** A callback for keyboard events, key is the ASCII value of the key pressed and x and y are the coordinates of the mouse at the moment of the keypress.

- **void glutMouseFunc(int button, int state, int x, int y);** A callback triggered on mouseclicks. Button can be GLUT_LEFT_BUTTON, GLUT_MIDDLE_BUTTON or GLUT_RIGHT_BUTTON. State defines what you actually did, did you press the button or release the button, these are defined by GLUT_UP or GLUT_DOWN. And x and y are the coordinates of the mouse.

- **void glutMotionFunc(int x, int y);** Callback triggered when the mouse is moved while one or more buttons are pressed. x and y are the coordinates.

Controls

- void glutMainLoop(void);
  - ◦ **Enters GLUT** event processing loop.
- **void glutPostRedisplay(void);**
  - ◦ that the function defined using glutDisplayFunc() will be called at the next opportunity. It marks the current window as needing a redraw.
- **void glutReshapeFunc(int width, int height);**
  - ◦ whenever the window is being moved and/or resized.
- **void glutTimerFunc(int msecs, func, value);**
  - ◦ registers the timer callback func to be triggered in at least msecs milliseconds.

Questions?

Further Reading

http://www.opengl.org/resources/libraries/glut/