RNS Institute of Technology
Department of CSE
Question Bank
Computer Graphics and Visualization (18CS62)

MODULE-1 and 5

- What is computer graphics?
- Applications of computer graphics.
- Different types of input devices.
- Definitions
  - Resolution
  - Pixel
  - Aspect ratio
  - Intensity
  - Horizontal and vertical deflection.
- Open GL related libraries and header files (library organization)
- Random and Raster scan approaches and differences.
- Explain video display devices – a. CRT b. Raster Scan c. Random Scan d. Flat panel
- Explain Screen, Relative and Absolute coordinate systems.
- DDA algorithm steps
- Advantages and disadvantages of DDA algorithm
- Bresenham's Line drawing algorithm.
- Explain Open GL line functions, line-attribute functions ,point function with examples.
- Derivation of the decision parameter by using Bresenham's method used to generate a straight line segment with slope<1 .
- Midpoint Algorithm Circle Algorithm
- Given a circle radius r(any value), at origin, using midpoint circle algorithm determine pixel position in first quadrant and plot the graph.
- Types of input devices with examples
- Logical Input Devices with examples
- **Lab program Bresenham's line**
- List the statement needed to setup an Open GL display window whose lowest-left corner is at pixel position (100,100) and with window aspect ratio 300X300.

Consider two raster systems with the resolutions of 640x480, 1280x1024, and 2560x2048. a) What size frame buffer (in butes) is needed for each of these systems to store 12 bits/pixel? How much storage is required for each system if 24 bits per pixel are to be stored?

Frame-buffer size for each of the systems is
640 × 480 × 12 bits ÷ 8 bits per byte = 450 KB
1280 × 1024 × 12 bits ÷ 8 bits per byte = 1920 KB
2560 × 2048 × 12 bits ÷ 8 bits per byte = 7680 KB
For 24 bits of storage per pixel, each of the above values is doubled.

Suppose an RGB raster system is to be designed using an 8 inch x 10 inch screen with a resolution of 100 pixels per inch n each direction. If we want to store 6 bits/pixel in the frame buffer, how much storage (in bytes) do we need for the frame buffer?

Storage needed for the frame buffer is
(8 inch x 100 pixels/inch) × (10 inch x 100 pixels/inch) × 6 bits ÷ 8 bits per byte ≈ 486 KB

Suppose we have a computer with 32 bits/word and a transfer rate of 1 million instructions/second (MIP). How long would it take to fill the frame buffer of a 300 dpi laser printer with a page size of 8.5 inches x 11 inches?

Total bits in the printer frame buffer is
8.5 × 11 × 300**2 ≈ 8.4 × 10**6 bits Therefore, loading time is
(8.4 × 10**6 bits) / (32 × 10**6 bps) ≈ 0.263 sec

How much time is spent scanning across each row of pixels during screen refresh on a raster system with a resolution of 1280 x 1024 and a refresh rate of 60 frames/second?

The scan rate for each pixel row is
60 frames/sec × 1024 lines/frame = 61, 440 lines/sec
And the scan time is approximately 16.3 microseconds per scan line. (Scan time per frame is 1/60 sec, or approximately 16.7 milliseconds.)

Explain the differences between the OpenGL core library, the OpenGL Utility, and the OpenGL Utility Toolkit?

The OpenGL core library contains hardware-independent functions, such as those for specifying primitives, attributes, geometric transformations, and three-dimensional viewing parameters. The GLU library contains functions for some other, more specialized operations, such as quadric-surface generation, B-spline surface generation, surface texture mapping, two-dimensional viewing, and some three-dimensional viewing operations. The GLUT library primarily provides hardware-dependent functions, such as those for display-window management and for interacting with input devices, but it also contains functions for generating various plane-surface, quadric-surface, and cubic-surface solids, such as a cube, sphere, cone, or teapot.

List the statements needed to set up an OpenGL display window whose lower-right corner is at pixel position (200, 200) with a window width of 100 pixels and a height of 75 pixels?

The GL Utility graphics calls include:

```
glutInitWindowPosition (100, 125);
glutInitWindowSize (100, 75);
```