LESSON 1 - Introduction to Computer Graphics

Computer Graphics

Computer Graphics is a broad field of computer science that focuses on the creation, modification, and display of visual data by computers.

They can be used to create anything from straightforward 2D shapes to intricate 3D models and animations, supporting a wide range of applications in various fields such as the entertainment, design, science, and educational sectors and others.

Three Main Tasks of Computer Graphics

1. **Modeling (*Shape*)**
   Creating and representing the geometry of objects in the 3D world.
2. **Rendering (*Light, Perspective*)**
   Generating 2D images of the objects.
3. **Animation (*Movement*)**
   Describing how objects change in time.

Key Concepts of Computer Graphics

1. **Rendering**
   The process of creating an image from a model (shown tremendous progress).

   Ray tracing has grown in popularity as a method of creating incredibly realistic graphics by mimicking the way light interacts with objects.

   Real-time ray tracing has been made possible by the introduction of GPUs with specialized ray tracing technology, such as NVIDIA's RTX series, which has improved virtual reality and gaming experiences.

2. **AI and Real-Time Graphics**
   Real-time rendering has changed as a result of AI being included into computer graphics.

   Real-time upscales of lower-resolution photos are achieved by AI techniques like Deep Learning Super Sampling (*DLSS*), which dramatically reduce processing load while preserving good visual quality. This is particularly crucial for VR and gaming, as keeping frame rates high is essential to the user experience.

3. **3D Modeling and Simulation**
   Over the last five years there have also been considerable advancements in 3D modeling. Programs such as Blender and Autodesk Maya have included more capable and user-friendly tools for 3D model creation.

   With the advancement of physics-based simulations, more realistic representations of materials and physical interactions are now possible, leading to increasingly precise and intricate animations in a variety of industries, including science visualization, video games, and movies.

4. **Augmented (*AR*) and Virtual (*VR*) Reality**
   Both AR and VR are becoming more and more common. VR and AR are now more widely available thanks to recent advancements in technology and software.

   Real-time tracking and rendering in conjunction with high-fidelity visuals have produced more dynamic and immersive environments utilized in professional training, education, and gaming.

5. **Digital Twins and the Metaverse**
   The idea of a communal virtual shared environment called the Metaverse has gained popularity. Businesses like Epic Games and Meta (*previously Facebook*) are making significant investments in building massive virtual worlds. The representation of expansive, dynamic three-dimensional worlds is one area where these settings mostly rely

on advancements in computer graphics. Advances in real-time 3D visualization and simulation have led to the emergence of digital twins, which are virtual reproductions of actual phenomena, as vital tools in engineering, healthcare, and urban planning.

Application Fields of Computer Graphics

1. **Graphical User Interface (GUI)**
2. **Arts and Advertising**
   Artificial modified image/s sequences.
3. **Visualizations**
   Graphs of functions, bar charts, and pie diagrams, temperature distribution on the surface of the earth, visualization of high-dimensional data, etc.
4. **Reconstructing 3D-objects from Measured Data**
   3D scanner, ultrasonic images, etc.
5. **CAD/CAM (*Computer-Aided Design or Manufacturing*)**
   For the design of objects like cars, buildings, etc.
6. **Simulation and Animation**
   Flight simulators, computer games, movies, etc.
7. **Interactive TV**
   Free choice of the viewer's position, computation of image based on information from a small number of cameras.
8. **Virtual Reality**
   Realistic 3D view + free movement + acoustics.
9. **Augmented Reality**
   Auxiliary information superimposed to the real world by semi-transparent glasses.

Real Scene to an Image

Real Scene whose details/objects have to be modeled.

Model of the Scene in which the objects of the real scene are represented by the available modeling techniques (*basic geometric objects, transformations, …*) The true geometry might only be approximated by the model.

Choice of a View/Part of the Virtual World to be displayed.

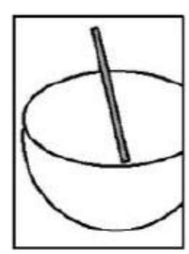
Clipping Computation which objects are within the chosen view

Visibility considerations: objects in the clipping region are visible to the viewer.

Result: pixel image.

Illumination effects, shading

Two-dimensional Clipping.



Legend:
Lesson | Header | Main Notes | Example | Extra Notes

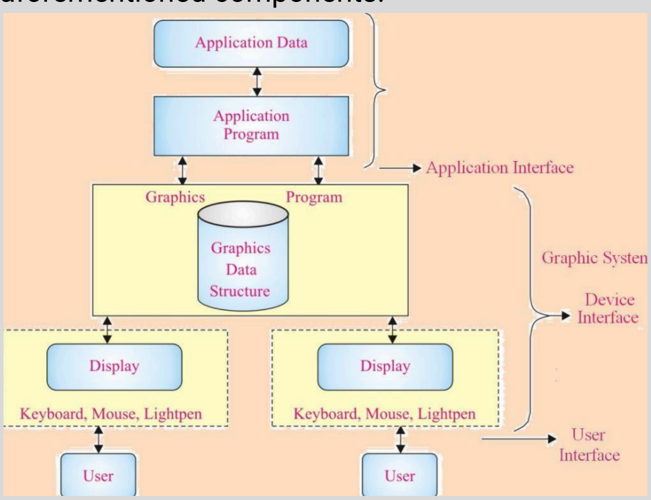
Graphics Systems Configuration

Graphic systems configuration describes how hardware and software are set up and optimized within a computer system to guarantee effective visual processing and rendering. For graphics-intensive applications like gaming, video editing, 3D rendering, and others, this is essential.

An outline of the essential components necessary in setting up a graphic system is as follows:

1. **Hardware Components**
2. **Software Components**
3. **Power Supply**
4. **Operating Systems and Software Updates**

The following sections are an in-depth look at the aforementioned components.



Hardware Components

Graphics Processing Unit (GPU)
The most critical component in a graphic system, responsible for rendering images, video, and animations. Modern GPUs are highly parallel processors optimized for handling complex calculations required in graphic rendering.

Configuration: Selecting the right GPU based on the workload (*e.g., gaming, professional 3D rendering, or AI processing*) is crucial. For gaming, high refresh rates and resolution support are important, while for rendering, VRAM and processing power are key.

Central Processing Unit (CPU)
While the GPU handles most of the graphic rendering tasks, the CPU is still crucial for processing game logic, AI, and other tasks that affect performance.

Configuration: A balanced CPU-GPU pairing is important to avoid bottlenecks. High core count and clock speed are beneficial for tasks that are both CPU and GPU-intensive.

Memory (RAM)
RAM is essential for handling the data being processed by both the CPU and GPU. More RAM allows for smoother multitasking and handling larger datasets in applications like 3D modeling or video editing.

Configuration: Typically, 16GB to 32GB of RAM is recommended for graphic-intensive tasks. Though more may be required for professional workloads

Storage
Fast storage options like SSDs are crucial for loading large textures, games, or video files quickly.

Configuration: NVMe SSDs are preferred for their high speed, which reduces load times in games and applications.

Monitor
The display resolution, refresh rate, and color accuracy of the monitor impact the visual experience.

Configuration: High refresh-rate monitors (*120Hz or higher*) are ideal for gaming, while 4K monitors with good color accuracy are preferred for video editing and graphic design.

Cooling Systems
Graphics-intensive tasks generate significant heat, so effective cooling (*both air and liquid*) is necessary to maintain performance and longevity of the hardware.

Configuration: High-performance fans, liquid cooling systems, and good airflow in the case are important for preventing thermal throttling.

Legend:
Lesson | Header | Main Notes | Example | Extra Notes

Software Components

Graphics Drivers
Drivers are software that enable communication between the operating system and the GPU.

Configuration: Keeping graphics drivers up to date is essential for optimizing performance and compatibility with the latest games and applications.

Graphics APIs
APIs like DirectX, Vulkan, and OpenGL provide a standardized way for software to interact with the GPU.

Configuration: Selecting the right API (*often determined by the application or game*) can improve performance. For instance, Vulkan and DirectX 12 are optimized for multi-threaded workloads.

Overclocking Software
Tools that allow users to increase the clock speed of the GPU and CPU to boost performance.

Configuration: Overclocking can provide performance boosts but should be done cautiously to avoid overheating and instability.

Graphics Settings
In applications and games, settings such as resolution, texture quality, anti-aliasing, and shadows can be adjusted to balance performance and visual quality.

Configuration: Tweaking these settings depending on the hardware capabilities can optimize the gaming or work experience. Lowering settings can improve frame rates, while higher settings enhance visual fidelity.

Power Supply

A sufficient power supply unit (PSU) is necessary to handle the power demands of the GPU, CPU, and other components.

Configuration: Ensuring that the PSU has enough wattage and proper efficiency (*like 80 Plus Gold or higher*) is crucial for stable system performance, especially when overclocking.

Operating Systems and Software Updates

The operating system must be configured to support the latest graphic technologies, with updates installed to ensure compatibility and performance.

Configuration: Using the latest OS version, with optimizations for gaming or graphic workstations, can enhance overall system stability and performance.

Graphic Systems

Three Components of Software

1. **Application Program**
   Creates, stores, and retrieves the data/objects to be pictured on the screen from *application data*.
2. **Application Data**
   Helps in producing images by sending a series or graphics output command in *graphics systems*.
3. **Graphics Systems**
   Interacts between the user and the application program and are responsible for producing the picture from the detailed descriptions and for passing the user input to the application program for processing.

The system consists of both hardware and software.

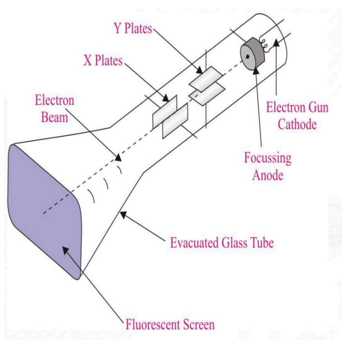
Legend:

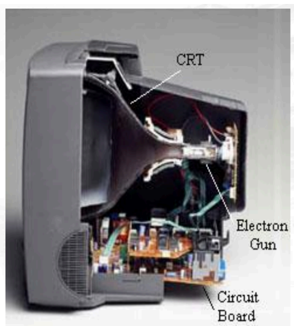
Lesson | Header | Main Notes | Example | Extra Notes

In computer graphics, the system's visual quality, performance, and energy efficiency are all greatly impacted by the display technology selected.

Over the years, four primary display technologies have been employed: *CRT (Cathode Ray Tube), LCD (Liquid Crystal Display), LED (Light-Emitting Diode), and plasma*. Each has pros and cons of its own.

1. CRT (Cathode Ray Tube)





Operation:
CRT displays work by firing electron beams from an electron gun towards a phosphorescent screen. The screen is coated with red, green, and blue phosphors that emit light when struck by the electrons, creating an image.

The electron beam scans the screen in a raster pattern, refreshing the image multiple times per second (*usually 60Hz or more*).

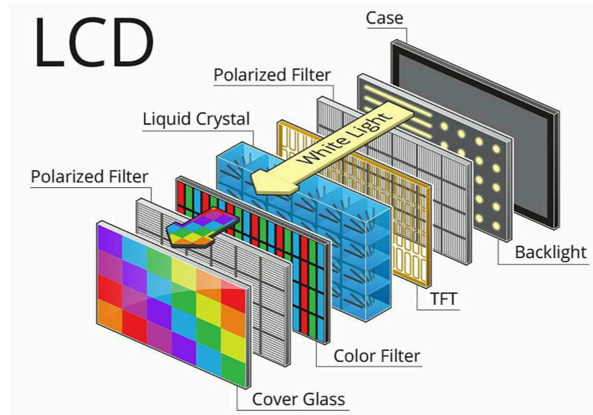
Characteristics:

a. **Color and Brightness:** CRTs offer excellent color reproduction and deep blacks, as they can individually control the intensity of each pixel.

b. **Response Time:** They have very fast response times, making them ideal for fast-moving images like in gaming.

c. **Drawbacks:** CRTs are large, heavy, and consume a lot of power. They also suffer from screen burn-in and are prone to flickering at lower refresh rates.

2. LCD (Liquid Crystal Display)



Operation:
LCDs use liquid crystals that do not emit light themselves but instead modulate light from a backlight. The liquid crystals are sandwiched between two layers of glass or plastic and are manipulated to block or allow light to pass through colored filters (red, green, and blue) to create an image.

The backlight is typically provided by CCFLs (Cold Cathode Fluorescent Lamps) in older LCDs.

Characteristics:

a. **Resolution:** LCDs have a fixed native resolution. Running at non-native resolutions can degrade image quality.

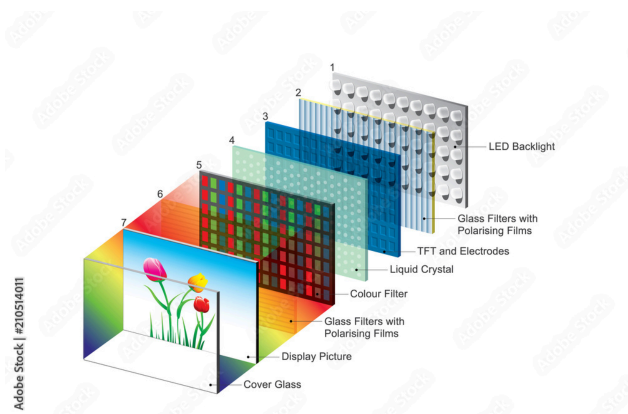
b. **Color and Brightness:** LCDs generally offer good color reproduction, but blacks may appear as dark gray due to the backlighting.

c. **Drawbacks:** Slower response times can lead to motion blur, and viewing angles are often limited.

3. LED (Light Emitting Diode)

Legend:

Lesson | Header | Main Notes | Example | Extra Notes



Operation:
LED displays are a type of LCD where the backlight is provided by LEDs instead of CCFLs,. LEDs can be either edge-lit or full-array, with some models featuring local dimming to improve contrast.

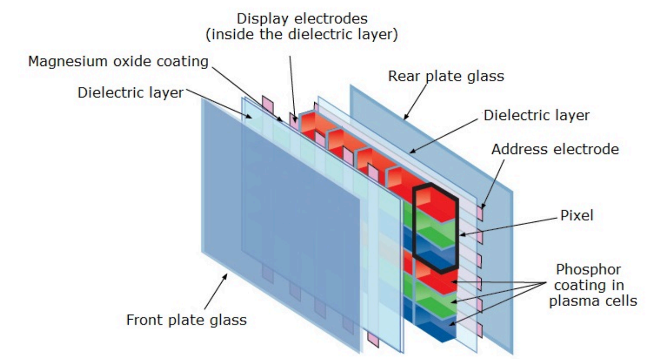
Characteristics:

a. **Brightness and Contrast:** LED displays offer better brightness and contrast ratios compared to traditional LCDs, especially with local dimming technology.

b. **Energy Efficiency:** LEDs are more energy efficient resulting in lower power consumption and heat generation.

c. **Drawbacks:** Like LCDs, they have a fixed native resolution and can suffer from limited viewing angles, though these have improved in newer models.

4. Plasma



Operation:
Plasma displays use small cells containing electrically charged ionized gases (plasma) to produce light. Each cell is essentially a tiny fluorescent lamp that emits UV light when excited by an electric current, which then excites phosphors to emit visible light.

Characteristics:

a. **Color and Brightness:** Plasma displays are known for deep blacks and good color accuracy, similar to CRTs. because they can control light emissions on a per-pixel basis.

b. **Viewing Angles:** They offer wide viewing angles with minimal color shift.

c. **Drawbacks:** Plasma displays consume more power, generate more heat, and are heavier compared to LCDs and LEDs. They are also susceptible to screen burn-in, especially with static images

Other Terms:
Thin-Film Transistor (TFT)

Raster Display

Legend:
Lesson | Header | Main Notes | Example | Extra Notes

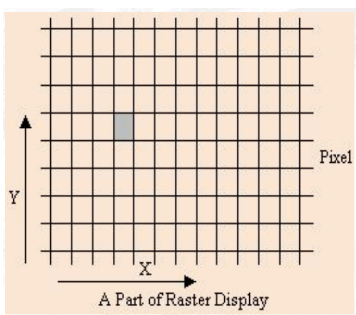
Raster Display
It is a type of screen or display system that renders images by illuminating pixels on a *grid* of individual points or dots, usually in a rectangular matrix.

Grid
Composed of rows and columns where each point on the grid, or pixel, can be individually controlled to display a specific color or brightness level.

Frame Buffer
A memory area in which a picture in the form of pixels is stored.

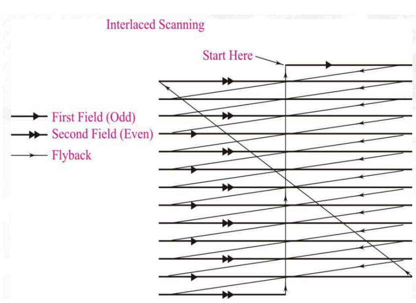
Raster Display Characteristics

1. Pixel-Based
The image is composed of thousands or millions of individual pixels. The quality of the image depends on the resolution, which is defined by the number of pixels in the grid.



A 512 x 512 (i.e. 2^9 x 2^9) element square raster requires 2^{18} or 262144 memory bits in a single-bit plane.

2. Scan Lines
The display refreshes the image by scanning the pixels row by row, from top to bottom. This is known as raster scan.



The screen is scanned from left to right, top to bottom all the time to generate graphics.

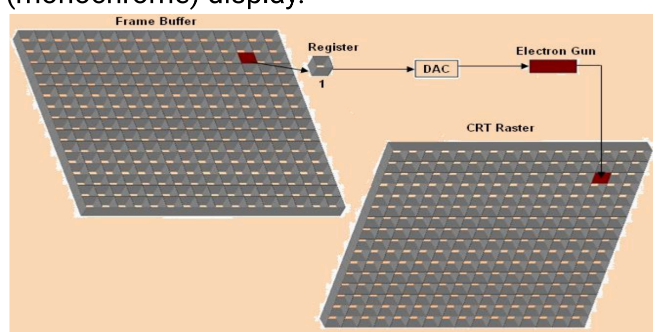
3. Color Depth
Each pixel can display a range of colors depending on the color depth, which is determined by the number of bits used to represent the color of a pixel.

4. Common in Screens
Raster displays are the most common type of display used in devices such as televisions, computer monitors, and smartphones.

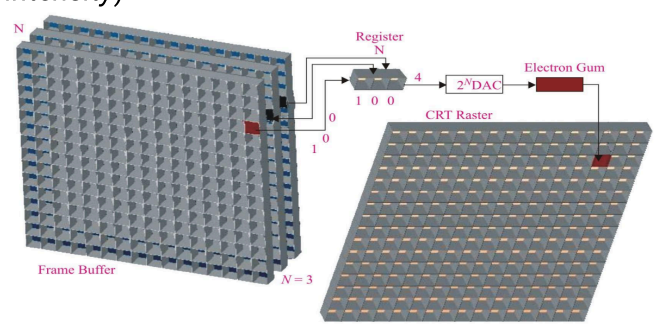
5. Resolution and Refresh Rate
The resolution defines how many pixels the display can show horizontally and vertically/ The refresh rate indicates how often the image is updated per second.

Screen must be refreshed or redrawn minimum 30 to 60 times per second (*30 to 60 hz*) to maintain flicker free image.

Information transfer from Frame Buffer to CRT
The picture is built up in the frame buffer one bit at a time (either 0 or 1) causing black or white (monochrome) display.



N-Bit plane frame buffer for Monochrome Displays
Intensity level between 0 (*dark*) and 2^N - 1 (*full intensity*)

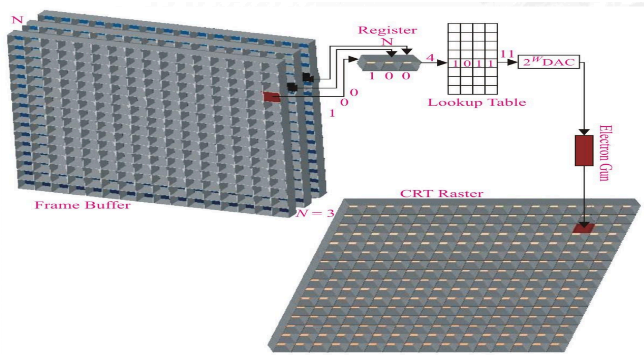


Increasing number of Intensity with Lookup Table

Legend:

Lesson | Header | Main Notes | Example | Extra Notes

Lookup tables contain 2^N entries for N-bit planes and with W width. Each entry will now have 2^W values.



Color Raster Display

Color frame buffer contains three bit planes for each primary color: red (R), green (G), and blue (B)

Each bit plane drives an individual color gun for each of the primary colors used in color video.

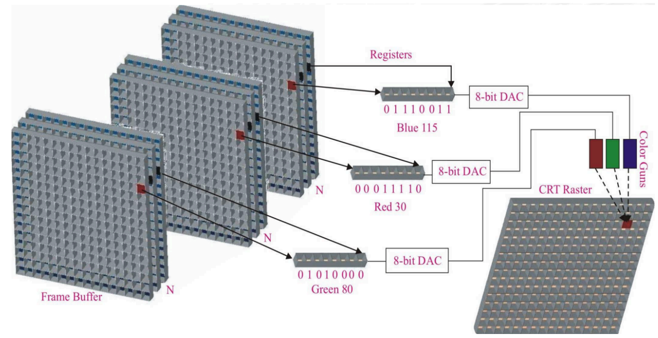
Table below shows a sample color table:

	Red	Green	Blue
Black	0	0	0
Red	1	0	0
Green	0	1	0
Blue	0	0	1
Yellow	1	1	0
Cyan	0	1	1
Magenta	1	0	1
White	1	1	1

Color Frame Buffer with Lookup Table

Increases the number of shades of each color

$(2^8)^3 = 2^{24} = 16{,}777{,}216$ possible colors.



Supporting Discipline

1. Computer science (algorithms, data structures, software engineering, …)
2. Mathematics (geometry, numerical, matrices…)
3. Physics (Optics, mechanics, …)
4. Psychology (Colour, perception)
5. Art and design

Legend:

Lesson | Header | Main Notes | Example | Extra Notes

LESSON 2 - Basic Principles of 2-Dimensional Graphics

Raster vs Vector Graphics

Vector Graphics is a representation by basic geometric objects (lines, circles, ellipses, cubic curves, …)

[insert image]

Raster Graphics is a representation in the form of pixel matrix.

[insert image]

d

Raster Graphics

Raster Graphics for Cathode Ray Tube
The video controller reads the images buffer (row-wise) from left to right and from right to left

At each pixel, the intensity of the rau os cjpsem accprdomg to the entry in the image buffer

60HZ rate blah blah blah

d

Vector Graphics

Scalable

Requires computations for display on a pixel-oriented medium (scan conversion)

Scan conversion can lead to ALIASING EFFECTS (such as jagged edges) which occur in general when a discrete sampling rate is used to measure a continuous??????

d

Raster as a Grid

Sometimes the y-coordinates are counted from top to bottom (from java)

Pixel lie on the grid points

Pixels with coordinates (5, 3)
[insert image]

d

Getting Started with Java 2D

AWT (Abstract Windowing Toolkit) components displayed on the screen have a paint method with Graphics objects as argument

The class Graphics2D within JAva 2D extends the class graphics

In order to exploit the options of java 2d the graphics object must be casted into a graphics 2d blah blah blah

d

d

Window Coordinates

Coordinates in the upper left corner is (0, 0)

E

d

d

Points used for the definition of the other objects (i.e. a line connecting two points).

Lines polylines or curves can be defined by two or more points

Areas are usually bounded by a closed polylines

Legend:

Lesson | Header | Main Notes | Example | Extra Notes

or polygons. Areas can be filled with color or a texture

d

Basic Geometric Objects

Line is a connecting line between two points

Polyline is a sequence of line where the following line starts where the previous one ends

Polygon closed

d

Polygons

Important Additional Properties of Polygons:

Non-self Overlapping
Vonvexity

A self-overlapping, non-convex and convex polygon

d

ParametricCurves

Parametric Curves
Quadratic Curves has 2 endpoints and 1 control point
Cubic Curves has 2 endpoints and 2 control points.

Parametric CUrve

Avoidinf sharp bends when attaching a cubic curve to a line

d

Gemoteric Objects in Java 2D

The abstract class Shape with it various subclasses, allows the construction of various two-dimensional geometric objects.
Bector graphics is used to define Shape BRO

Shapes will not be drawn until the draw or the fill method is called with the corresponding Shape as argument in the form

graphics2d.draw (shape)
graphics2d.fill (shape)

The abstract class Point2D is not a subclass of Shape
Points cannot be drawn in Java.They are only supposed to be used for the description of other objects.
Subclass of Point2D;Point2D.Float and Point2D. Double.

d

d

Line (Segment): two endpoints are needed to define a line

Line2D.Double line = new Line2D.Double (x1,y,1,x2,y2)

QuadCurve2D.Double qc = new QuadCurve2D.Double(x1,y1,ctrlx,ctrly,x2,y2);

d

d

d

d

d

d

Legend:

Lesson | Header | Main Notes | Example | Extra Notes

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

Legend:

Lesson | Header | Main Notes | Example | Extra Notes

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

d

v

d

d

d