KARPAGAM ACADEMY OF HIGHER EDUCATION

(Deemed University Established Under Section 3 of UGC Act 1956)

Coimbatore - 641021.

(For the candidates admitted from 2015 onwards)

DEPARTMENT OF COMPUTER SCIENCE, CA & IT

SUBJECT : **COMPUTER GRAPHICS**
SEMESTER : **V**

SUBJECT CODE : **15CSU503** **CLASS** : **III B.Sc.CS**

Course Objective: This course presents an introduction to computer graphics designed to give the student an overview of fundamental principles. The course will include an overview of common graphics hardware, 2D and 3D transformations and viewing, and basic raster graphics concepts such as scan-conversion, and clipping. Methods for modeling objects as polygonal meshes or smooth surfaces, and as rendering such as hidden-surface removal, shading, illumination, and shadows will be investigated.

Course Learning Outcomes:

A student who successfully completes this course should be able to:

- have a knowledge and understanding of the structure of an interactive computer graphics system, and the separation of system components.
- have a knowledge and understanding of geometrical transformations and 3D viewing.
- create interactive graphics applications.
- have a knowledge and understanding of techniques for representing 3D geometrical objects.
- have a knowledge and understanding of the fundamental principles of local and global illumination models.

UNIT-I

A Survey of Computer Graphics- Video Display Devices- Refresh Cathode-Ray Tubes-Raster Scan Displays-Random Scan Displays-Color CRT Monitors-Direct –View Storage Tubes-Flat Panel Displays-Raster Scan Systems-Three Dimensional Viewing Devices-Random Scan Systems.

UNIT-II

Input Devices: Keyboards-Mouse –Track Ball and Space ball-Joysticks-Data Glove- digitizers- Image Scanners- Touch Panels-Light Pens-Voice Systems-**Hard Copy Devices**: Printers and Plotters

UNIT-III

Point and Lines- Line Drawing Algorithms: DDA Algorithm- Bresenhams Line Algorithm. **Circle Generating Algorithms**: Mid Point Circle Algorithm. Two Dimensional Geometric Transformations: **Basic Transformations**: Translation-Rotation-Scaling-**Composite Transformations**: Translations-Rotations- Scaling. General Pivot Point Rotation- General Fixed Point Scaling.

UNIT-IV

Two Dimensional Viewing: The Viewing Pipeline- Window to view port Transformation- **Clipping Operations**-Point Clipping -Line Clipping: Cohen Sutherland Line Clipping. Polygon Clipping: Sutherland –Hodgeman Polygon Clipping-Text Clipping.

UNIT-V

Three Dimensional Display Methods-Parallel Projection- three Dimensional Geometric Transformations: Translation-Rotations- Scaling. **Projections**: Parallel Projections- Perspective Projections. **Visible Surface Detection Methods**: Classification of Visible Surface Detection Algorithms-Back Face Detection-Depth Buffer Method- Depth Buffer Method- Area Sub division Method.

TEXT BOOK

1. Donald Hearn, M. Pauline Baker., 2007. *Computer Graphics-C Version*, 2nd Edition, Pearson Education, New Delhi.
   (Page Nos. : 24-54, 56-77, 80-92, 103-118, 204-215, 236-256, 427-443, 458-463, 490-495, 502-505)

REFERENCES

1. Amarendra N. Sinha. 2008. *Computer Graphics*, 1st Edition ,Tata McGraw Hill, New Delhi.
2. Foley, Vandam, Feiner and Hughes. 1999. *Computer Graphics Principles and Practices*, 2nd Edition ,Addison Wesley, Singapore.
3. Evangeline D., Anita, S. 2016. *Computer Graphics And Multimedia Insights, Mathematical Models And Programming Paradigms*. PHI Learning Pvt. Ltd.

4. Rae Earnshaw.2014.*Computer Graphics: Developments in Virtual Environments*, 2nd Edition ,academic Press, New Delhi.

WEBSITES:

1. www.cgshelf.com
2. www.cg**tutorials**.com
3. www.allgraphicdesign.com

KARPAGAM ACADEMY OF HIGHER EDUCATION

(Deemed to be University Established Under Section 3 of UGC Act 1956)

Pollachi Main Road, Eachanari Post, Coimbatore – 641 021. INDIA

Phone : 0422-6471113-5, 6453777 Fax No : 0422 -2980022-3

Email : info@karpagam.com Web : www.kahedu.edu.in

LECTURE PLAN
DEPARTMENT OF COMPUTER SCIENCE, CA & IT

STAFF NAME : Dr.P.TAMIL SELVAN
SUBJECT NAME : COMPUTER GRAPHICS SUB.CODE:15CSU503
SEMESTER : V CLASS: III B.Sc (CS)

S.No.	Lecture Duration Hour	Topics to be Covered	Support Material/Page Nos
		UNIT-I	
1	1	Introduction to Computer Graphics	T1:17
2	1	A Survey of Computer Graphics	T1:22
3	1	Video display devices	T1:56
4	1	Refresh cathode-ray tubes	T1:57-58 W1
5	1	Electrostatic deflection beam in CRT	T1:59-60
6	1	Raster scan displays Random scan displays	T1:60-62,R1-47
7	1	Color CRT monitors • Beam penetration method	T1:62-63
8	1	Color CRT monitors • Shadow mask method	T1:63-64
9	1	Direct-view storage tubes	T1:65
10	1	Flat panel displays • Emissive displays	T1:65-66
11	1	Flat panel displays • Non-emissive displays	T1:66-68
12	1	Raster scan systems	T1:73
13	1	Three dimensional viewing devices	T1:69-72
14	1	Random scan systems	T1:76-77

15	1	Recapitulation and Discussion of Important Questions	
Text Book Reference Book Websites	T1: Donald Hearn and M. Pauline Baker. 2007. *Computer Graphics-C Version*, 2nd Edition, Pearson Education, New Delhi. R1: Amarendra N. Sinha. 2008. *Computer Graphics*,1st Edition,Tata McGraw Hill,New Delhi. W1: www.openlearning.intercol.edu/Computer graphics.		
	Total No of Hours Planned For Unit 1=15		
		UNIT-II	
1	1	Introduction to Input devices	T1:80, W1
2	1	Keyboard, Mouse	T1:81-82
3	1	Track ball	T1:82-83
4	1	Space ball, Joysticks	T1:83-84, w1
5	1	Data glove	R1:155-157
6	1	Digitizers, Image scanners	T1:84-87,R1
7	1	Touch panels	T1:88-89
8	1	Optical touch panel	W3
9	1	Electrical touch panel, Light pens	T1:89-90
10	1	Voice systems	T1:90-91
11	1	Introduction to Hard-copy devices	T1:92-94 R1:12-15
12	1	Printers	T1:95-98
13	1	Plotters	w1
14	1	Recapitulation and Discussion of Important Questions	
Text Book Reference Book	T1: Donald Hearn and M. Pauline Baker. 2007. *Computer Graphics-C Version*, 2nd Edition, Pearson Education, New Delhi. R1: Amarendra N. Sinha. 2008. *Computer Graphics*,1st Edition,Tata McGraw Hill,New Delhi. W1: www.openlearning.intercol.edu/Computer graphics		

Websites	W3: https://www.touchsystems.com/opticaltouch		
	Total No of Hours Planned For Unit II=14		
		UNIT-III	
1	1	Points and lines	T1:104-106
2	1	Line Drawing Algorithm	T1:106-107 R1:20-25
3	1	DDA Algorithm	T1:107-108
4	1	Bresenham's Line Algorithm	T1:108-110 R1:25-27
5	1	Circle Generating Algorithms	T1:117
6	1	Properties of Circles	T1:118
7	1	Midpoint Circle Algorithm	T1:118-121
8	1	2 Dimensional Geometric Basic Transformation	T1:204
9	1	Translation	T1:204-205, w2
10	1	Rotations, Scalings	T1:206-208
11	1	Composite Transformation-Translations, Rotations	T1:209-211
12	1	Scalings	T1: 211-212
13	1	General Pivot Point Rotation	T1:212-213
14	1	General Fixed Point Scaling	T1: 212
15	1	Recapitulation and Discussion of Important Questions	
Text Book Reference Book Websites	T1: Donald Hearn and M. Pauline Baker. 2007. *Computer Graphics-C Version,* 2nd Edition, Pearson Education, New Delhi. R1: Amarendra N. Sinha. 2008. *Computer Graphics*,1st Edition,Tata McGraw Hill,New Delhi. W2: http://www.cgtutorials.com.		
	Total No of Hours Planned For Unit III=15		

		UNIT-IV	
1	1	Introduction to Two Dimensional Viewing	T1:236
2	1	Viewing Pipeline	T1:237
3	1	The 2 Dimensional viewing transformation pipeline	T1:238-240,W2
4	1	Window to View Port Transformation	T1:240-242 R2:122
5	1	Clipping Operations	T1:244-245 W2
6	1	Point Clipping, Line Clipping	T1:245 R2:123
7	1	Cohen Sutherland Line Clipping	T1:246-248
8	1	Liang-Barsky Line Clipping	T1:250-251
9	1	Polygon Clipping	T1:257 R2:128-130
10	1	Sutherland- Hodgeman Polygon Clipping	T1:258-262
11	1	Types of Clipping • Text Clipping	T1:264
12	1	• Curve Clipping	T1:265
13	1	• Polygon Clipping	T1:266
14	1	• Line Clipping	T1:267
15	1	Recapitulation and Discussion of important Questions	
Text Book Reference Book Websites	T1: Donald Hearn and M. Pauline Baker. 2007. Computer Graphics-C Version, 2nd Edition, Pearson Education, New Delhi. R2: Foley, Vandam, Feiner and Hughes. 1999. *Computer Graphics Principles and Practices*, 2nd Edition, Addison Wesley, Singapore. W2: http://www.cgtutorials.com		
	Total No of Hours Planned For Unit IV=15		
		UNIT-V	
1	1	Three dimensional display methods	T1:317-318 W1
2	1	Parallel projection	T1:318-319
3	1	Three Dimensional Geometric Transformations	T1:427-428

4	1	Translation	T1:428-429
5	1	Rotation-Coordinate axes rotation	T1:429-432
6	1	Scaling	T1:440-441
7	1	Parallel projection	T1:459,460
8	1	Perspective projection	T1:463-466,R2
9	1	Classification of visible surface detection algorithms	T1:490
10	1	Back face detection	T1:491-492
11	1	Depth Buffer Method	T1:492-495,w2
12	1	Area sub division method	T1:502-503
13	1	Recapitulation and Discussion of important Questions	
14	1	Discussion of Previous ESE Question Papers.	
15	1	Discussion of Previous ESE Question Papers.	
16	1	Discussion of Previous ESE Question Papers.	
Text Book Reference Book Websites	T1: Donald Hearn and M. Pauline Baker. 2007. Computer Graphics-C Version, 2nd Edition, Pearson Education, New Delhi. R2: Foley, Vandam, Feiner and Hughes. 1999. *Computer Graphics Principles and Practices*, 2nd Edition, Addison Wesley, Singapore. W1: www.openlearning.intercol.edu/Computer graphics W2: http://www.cgtutorials.com		
	Total No of Hours Planned for unit V=16		
Total Planned Hours	**75**		

TEXT BOOK

1. Donald Hearn and M. Pauline Baker. 2007. *Computer Graphics-C Version*, Pearson Education, New Delhi, 2nd Edition.

   (Page Nos. : 24-54, 56-77, 80-92, 103-118, 204-215, 236-256, 427-443, 458-463, 490-495, 502-505)

REFERENCES

1. Amarendra N. Sinha. 2008. *Computer Graphics*, 1st Edition, Tata McGraw Hill, New Delhi.
2. Foley, Vandam, Feiner and Hughes. 1999. *Computer Graphics Principles and Practices*, 2nd Edition, Addison Wesley, Singapore.
3. Evangeline D., Anita, S. 2016. *Computer Graphics and Multimedia Insights, Mathematical Models and Programming Paradigms*. PHI Learning Pvt. Ltd.
4. Rae Earnshaw.2014.*Computer Graphics: Developments in Virtual Environments*, 2nd Edition, academic Press, New Delhi.

WEBSITES

W1: www.openlearning.intercol.edu/Computer graphics

W2: http://www.cgtutorials.com

W3: https://www.touchsystems.com/opticaltouch

KARPAGAM ACADEMY OF HIGHER EDUCATION

(Deemed University Established Under Section 3 of UGC Act 1956)
Coimbatore - 641021.
(For the candidates admitted from 2015 onwards)

DEPARTMENT OF COMPUTER SCIENCE, CA & IT

SUBJECT : COMPUTER GRAPHICS
SEMESTER : V
SUBJECT CODE: 15CSU503 CLASS : III B.Sc.CS

UNIT-I

A Survey of Computer Graphics- Video Display Devices- Refresh cathode-Ray Tubes-Raster Scan Displays-Random Scan Displays-Color CRT Monitors-Direct –View Storage Tubes-Flat Panel Displays-Raster Scan Systems-Three Dimensional Viewing Devices-Random Scan Systems.

TEXT BOOK

1. Donald Hearn and M. Pauline Baker. 2007. Computer Graphics-C Version, 2nd Edition, Pearson Education, New Delhi.
   (Page Nos. : 24-54, 56-77, 80-92, 103-118, 204-215, 236-256, 427-443, 458-463, 490-495, 502-505)

REFERENCES

1. Amarendra N. Sinha. 2008. *Computer Graphics*, 1st Edition ,Tata McGraw Hill, New Delhi.
2. Foley, Vandam, Feiner and Hughes. 1999. *Computer Graphics Principles and Practices*, 2nd Edition ,Addison Wesley, Singapore.
3. Evangeline D., Anita, S. 2016. *Computer Graphics and Multimedia Insights, Mathematical Models and Programming Paradigms*. PHI Learning Pvt. Ltd.
4. Rae Earnshaw.2014.*Computer Graphics: Developments in Virtual Environments*, 2nd Edition, academic Press, New Delhi.

WEB SITES:

1. www.cgshelf.com
2. www.cgtutorials.com
3. www.allgraphicdesign.com

UNIT I

A SURVEY OF COMPUTER GRAPHICS

Definition:

Computer Graphics is the discipline of producing picture or images using a computer which include modeling, creation, manipulation, storage of geometric objects, rendering, converting a scene to an image, the process of transformations, rasterization, shading, illumination, animation of the image, etc.

Uses of Computer Graphics:

Computer Graphics has been widely used in graphics presentation, paint systems, computer-aided design (CAD), image processing, simulation, etc. From the earliest text character images of a non-graphic mainframe computers to the latest photographic quality images of a high resolution personal computers, from vector displays to raster displays, from 2D input, to 3D input and beyond, computer graphics has gone through its short, rapid changing history.

From games to virtual reality, to 3D active desktops, from unobtrusive immersive home environments, to scientific and business, computer graphics technology has touched almost every concern of our life. Before we get into the details, we have a short tour through the history of computer graphics

A SURVEY OF COMPUTER GRAPHICS

Computers have become a powerful tool for the rapid and economical production of pictures. There is virtually no area in which graphical displays cannot be used to some advantage, and so it is not surprising to find the use of computer graphics so widespread.

Although early applications in engineering and science had to rely on expensive and cumbersome equipment, advances in computer technology have made interactive computer graphics a practical tool. Today, we find computer graphics used routinely in such diverse areas as science, engineering, medicine, business, industry, government, art, entertainment, advertising, education, and training.

COMPUTER-AIDED DESIGN

A major use of computer graphics is in design processes, particularly for engineering and architectural systems, but almost all products are now computer designed. Generally referred to as CAD, computer-aided design methods are now routinely used in the design of buildings, automobiles, aircraft, watercraft, spacecraft, computers, textiles, and many, many other products.

Software packages for CAD applications typically provide the designer with a multi-window environment. Circuits such as the one shown in below Figure and networks for communications, water supply, or other utilities are constructed with repeated placement of a few graphical shapes.

The shapes used in a design represent the different network or circuit components. Standard shapes for electrical, electronic, and logic Circuits are often supplied by the design package. For other applications, a designer can create personalized symbols that are to be used to construct the network or circuit.

The system is then designed by successively placing components into the layout, with the graphics package automatically providing the connections between components. This allows the designer t~ quickly try out alternate circuit schematics for minimizing the number of components or the space required for the system.

Fig: A circuit design application using

Animations are often used in CAD applications. Real-time animations using wire frame displays on a video monitor are useful for testing performance of a vehicle or system, when we do not display objects with rendered surfaces, the calculations for each segment of the animation can be performed quickly to produce a smooth real-time motion on the screen. Also, wire frame displays allow the designer to see into the interior of the vehicle and to watch the behavior of inner components during motion.

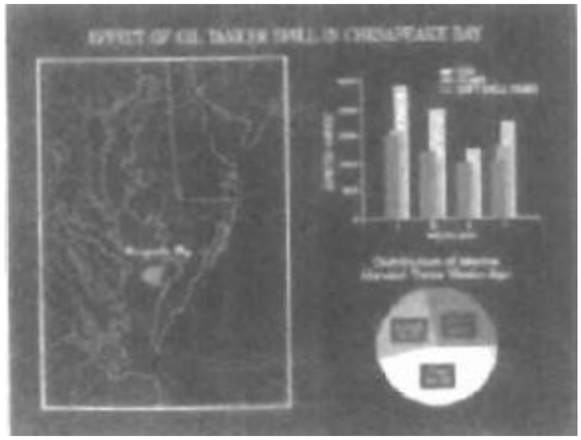
Animations in virtual reality environments are used to determine how vehicle operators are affected by certain motions.

This allows the designer to explore various positions of the bucket or backhoe that might obstruct the operator's view, which can then be taken into account in the overall hector design.

PRESENTATION GRAPHICS

Another major application area is presentation graphics, used to produce illustrations for reports or to generate 35-mm slides or transparencies for use with projectors. Presentation graphics is commonly used to summarize financial, statistical, Mathematical, scientific, and economic data for research reports, managerial reports, consumer information bulletins, and other types of reports.

Workstation devices and service bureaus exist for converting screen displays into 35-mm slides or overhead transparencies for use in presentations. Typical examples of Presentation graphics are bar charts, line graphs, surface graphs, pie charts, and other displays showing relationships between multiple parameters.



(Fig: Two dimensional bar chart and pie chart)

COMPUTER ART

Computer graphics methods are widely used in both fine art and commercial art applications. Artists use a variety of computer methods, including special-purpose hardware, artist's paintbrush (such as Lumens), other paint packages (such as Pixel paint and Super paint), specially developed software, symbolic mathematics packages (such as Mathematics), CAD packages, desktop publishing software, and animation packages that provide facilities for designing object shapes and specifying object motions.

Fine artists use a variety of other computer technologies to produce images. For many applications of commercial art (and in motion pictures and other applications), photo realistic techniques are used to render images of a product.

A common graphics method employed in many commercials is morphing, where one object is transformed (metamorphosed) into another. This method has been used in TV commercials to turn an oil can into an automobile engine, an automobile into a tiger, a puddle of water into a tiger, and one person's face into another face

(Fig: Cartoon drawing produced with a paintbrush program)

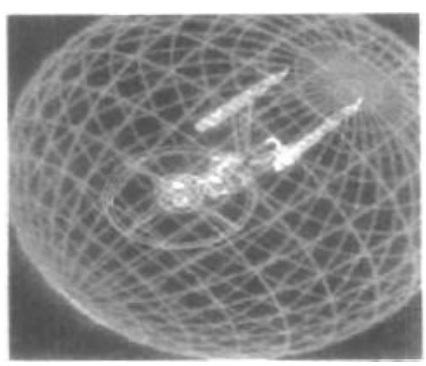
In above figure illustrates the basic idea behind a paintbrush program that allows artists to "paint" pictures on the screen of a video monitor.

Actually, the picture is usually painted electronically on a graphics tablet (digitizer) using a stylus, which can simulate different brush strokes, brush widths, and colors. A paintbrush program was used to create the characters.

ENTERTAINMENT

Computer graphics methods are now commonly used in making motion pictures, music videos, and television shows. Sometimes the graphics scenes are displayed by themselves.

- A graphics scene generated for the movie Star Trek-the Wrath of Khan is shown in below figure.



(Fig: Graphics developed for the paramount picture movie)

- The planet and spaceship are drawn in wire fame form and will be shaded with rendering methods to produce solid surfaces. Many TV series regularly employ computer graphics methods.
- Music videos use graphics in several ways.

Graphics objects can be combined with the live action, or graphics and image processing techniques can be used to produce a transformation of one person or object into another (morphing).

An example of morphing is shown in the sequence of scenes in below Figure, produced for the David Byme video She's Mad.

(Examples of morphing)

EDUCATION AND TRAINING

Computer-generated models of physical, financial, and economic systems are often used as educational aids. Models of physical systems, physiological systems, population trends, or equipment, such as the color coded diagram in below figure, can help trainees to understand the operation of the system.

For some training applications, special systems are designed. Examples of Such specialized systems are the simulators for practice sessions or training of ship captains, aircraft pilots, heavy-equipment operators, and air traffic control personnel.

Some simulators have no video screens**;** for example, a flight simulator with only a control panel for instrument flying. But most simulators provide graphics screens for visual operation.

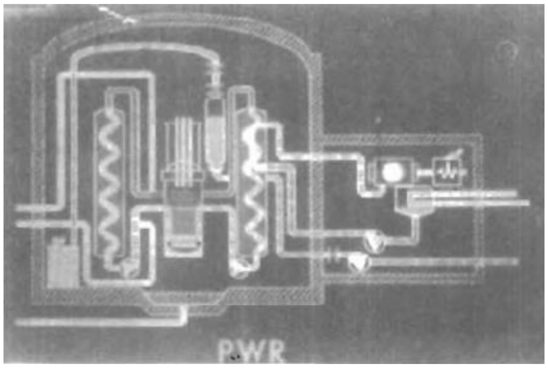


Fig: color coded diagram

IMAGE PROCESSING:-

Although methods used in computer graphics and Image processing overlap, the two areas with fundamentally different operations. In computer graphics, a computer is used to create a picture. Image processing, on the other hand.

Applies techniques to modify or interpret existing picture, such as photographs and TV scans.

Two principal applications of image processing are

(1) Improving picture quality

(2) Machine perception of visual information,

It is used in robotics. To apply image processing methods, we first digitize a photograph or other Picture into an image file. Then digital methods can be applied to rearrange picture parts, to enhance color separations, or to improve the quality of shading.

Example of the application of image processing methods to enhance the quality of a picture is shown

These techniques are used extensively in commercial art applications that involve the retouching and rearranging of sections of photographs and other artwork.

- Similar methods are used to analyze satellite photos of the earth and photos of galaxies.

- Medical applications also make extensive use of image processing techniques for picture enhancements, in tomography and in simulations of operations.

Tomography is a technique of X-ray photography that allows cross-sectional views of physiological systems to be displayed. Both computed X-rav tomography (CT) and position emission tomography (PET) use projection methods to reconstruct cross sections from digital data.

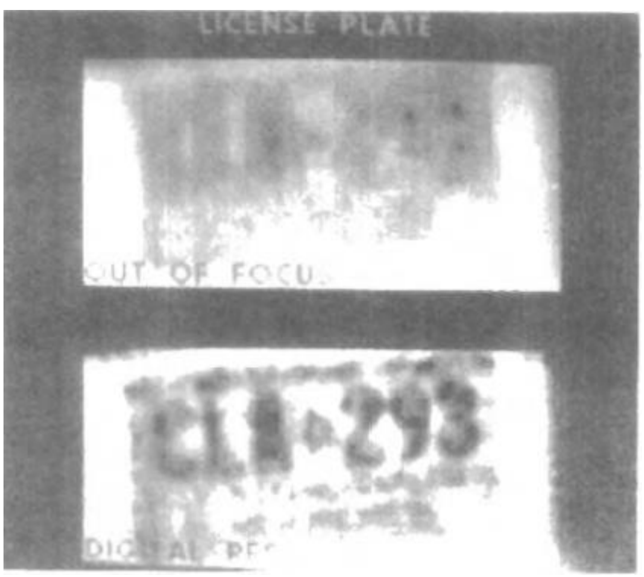


Fig. (The blurred photograph of a license plate)

GRAPHICAL USER INTERFACES (GUI):-

It is common now for software packages to provide a graphical interface. A major component of a graphical interface is a window manager that allows a user to display multiple-window areas.

Each window can contain a different process that can contain graphical or no graphical displays. To make a particular window active, we simply click in that window using an interactive pointing device. Interfaces also display menus and icons for fast selection of processing options or parameter values.

An icon is a graphical symbol that is designed to look Like the processing option it represents. The advantages of icons are that they take up less screen space than corresponding textual descriptions and they can be understood more quickly if well designed. Menus contain lists of textual descriptions and icons.

- **Multiple window areas**
- **Menus and icons**

Fig: A Graphical user interface showing

In above figure illustrates a typical graphical interface, containing a window manager, menu displays, and icons.

Example the menus allow selection of processing options, color values, and graphics parameters. The icons represent options for painting, drawing, zooming, typing text strings, and other operations connected with picture construction.

VIDEO DISPLAY DEVICES :-

The primary output device in a graphics system is a video monitor (Fig**).** The operation of most video monitors is based on the standard cathode-ray tube (CRT) design, but several other technologies exist and solid-state monitors may eventually predominate.

Fig: (A computer graphics workstation)

REFRESH CATHODE-RAY TUBES

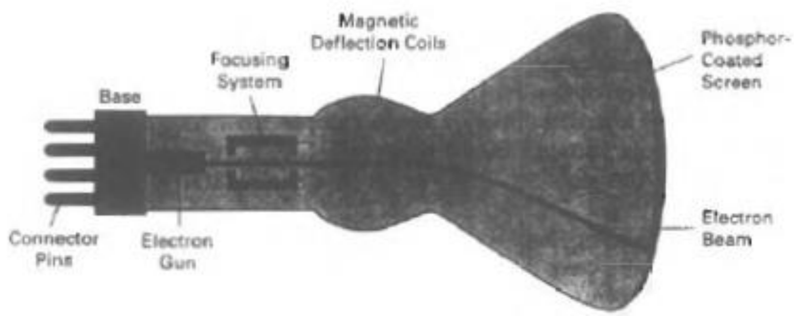
The basic operation of a CRT. A beam of electrons (cathode rays) emitted by an electron gun, passes through focusing and deflection systems that direct the beam toward specified positions on the phosphor coated screen.

The phosphor then emits a small spot of light at each position contacted by the electron beam. Because the light emitted by the phosphor fades very rapidly, some method is needed for maintaining the screen picture.

One way to keep the phosphor glowing is to redraw the picture repeatedly by quickly directing the electron beam back over the same points. This type of display is called a refresh CRT.

The primary components of an electron gun in a CRT are the heated metal cathode and a control grid which is shown in below Figure. Heat is supplied to the cathode by directing a current through a coil of wire, called the filament, inside the cylindrical cathode structure.

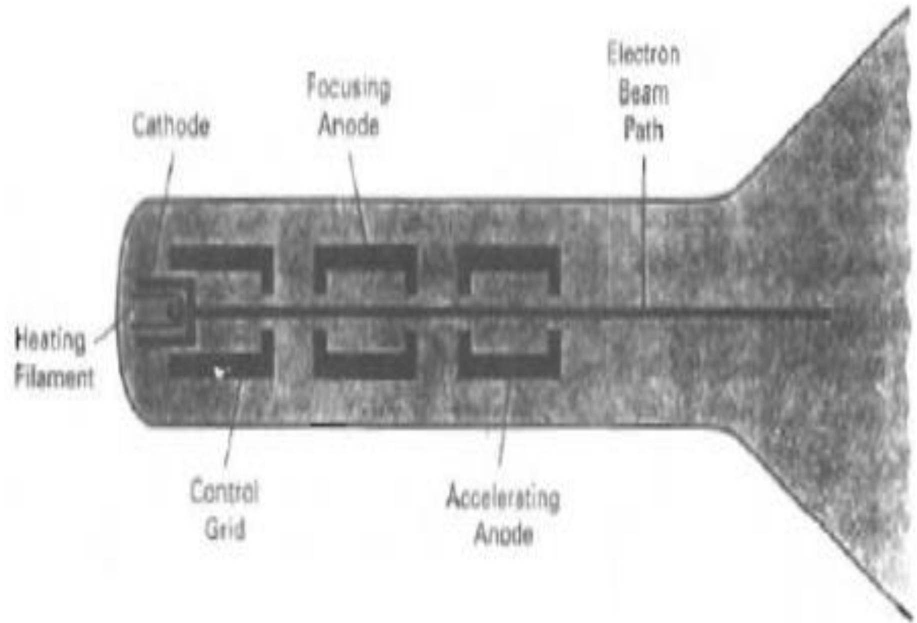
This causes electrons to be 'boiled off" the hot cathode surface. In the vacuum inside the CRT envelope, the free, negatively charged electrons are then accelerated toward the phosphor coating by a high positive voltage.



(Fig: basic design of magnetic deflection CRT)

The accelerating voltage can be generated with a positively charged metal coating on the inside of the CRT envelope near the phosphor screen, or an accelerating anode can be used,

Sometimes the electron gun is built to contain the accelerating anode and focusing system within the same unit.



(Fig: Operation of the electron gun with an accelerating anode)

Intensity of the electron beam is controlled by setting voltage levels on the control grid, which is a metal cylinder that fits over the cathode.

A high negative voltage applied to the control grid will shut off the beam by repelling electrons and stopping them from passing through the small hole at the end of the control grid structure.

A smaller negative voltage on the control grid simply decreases the number of electrons passing through. Since the amount of light emitted by the phosphor coating depends on the number of electrons striking the screen, we control the brightness of a display by varying the voltage on the control grid.

The focusing system in a CRT is needed to force the electron beam to converge into a small spot as it strikes the phosphor. Otherwise, the electrons would repel each other, and the beam would spread out as it approaches the screen.

Focusing is accomplished with either electric or magnetic fields. Electrostatic focusing is commonly used in television and computer graphics monitors. With electrostatic focusing, the electron beam passes through a positively charged metal cylinder that forms an electrostatic lens.

The action of the electrostatic lens focuses the electron beam at the center of the screen, in exactly the same way that an optical lens focuses a beam of light at a particular focal distance.

Similar lens focusing effects can be accomplished with a magnetic field set up by a coil mounted around the outside of the CRT envelope. Magnetic lens focusing produces the smallest spot size on the screen and is used in special purpose Devices.

Additional focusing hardware is used in high-precision systems to keep the beam in focus at all screen positions. The distance that the electron beam must travel to different points on the screen varies because the radius of curvature for most CRTs is greater than the distance from the focusing system to the screen center.

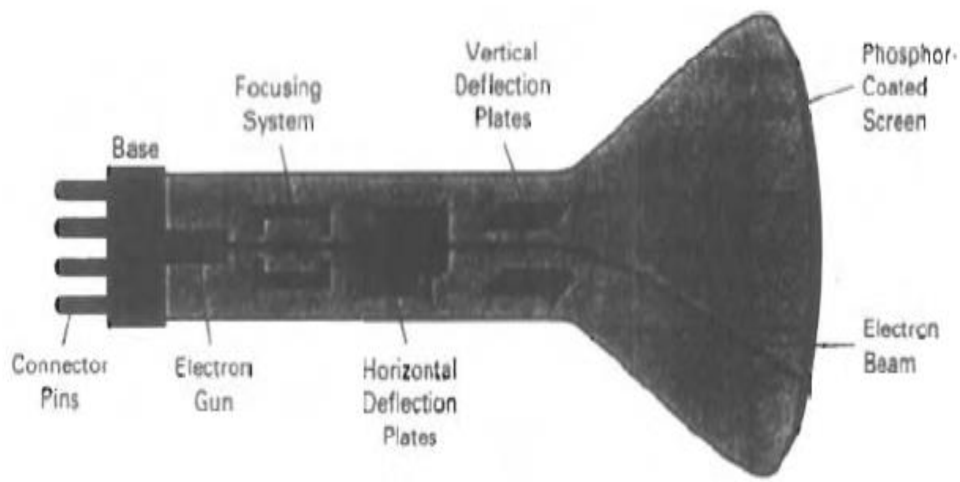
Cathode-ray tubes are now common. Constructed with magnetic deflection coils mounted on the outside of the CRT envelope.

Two pairs of coils are used, with the coils in each pair mounted on opposite sides of the neck of the CRT envelope.

- **One pair is mounted on the top and bottom of the neck and**
- **The other pair is mounted on opposite sides of the neck**

The magnetic, field produced by each pair of coils results in a transverse deflection force that is perpendicular both to the direction of the magnetic field and to the direction of travel of the electron beam. Horizontal deflection is accomplished with one pair of coils, and vertical deflection by the other pair.

The proper deflection amounts are attained by adjusting the current through the coils. When electrostatic deflection is used, two pairs of parallel plates are mounted inside the CRT envelope. One pair of plates is mounted horizontally to control the vertical deflection, and the other pair is mounted vertical to control horizontal deflection.



(Fig: Electrostatic deflection of the electron beam in a CRT)

Spots of light are produced on the screen by the transfer of the CRT.

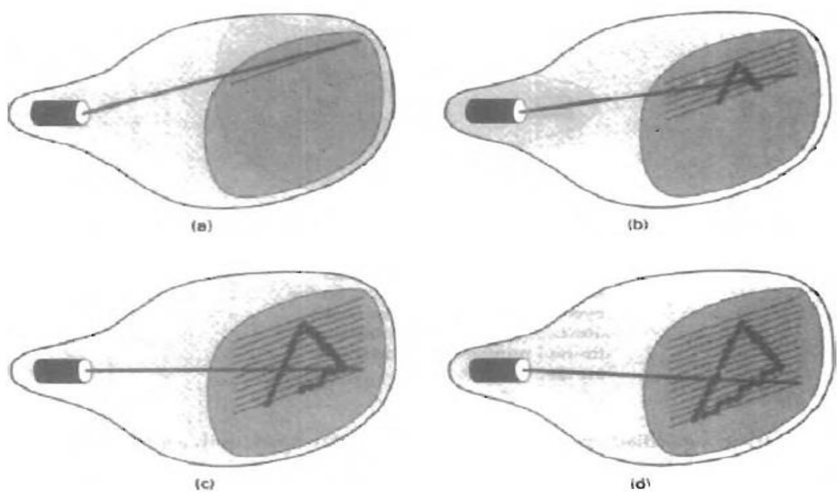
Remainder causes electrons in the phosphor atoms to move up to higher quantum-energy levels. After a short time, the "excited phosphor electrons begin dropping back to their stable ground state, giving up their extra energy as small quantum of Light energy.

The electron light emissions: a glowing spot that quickly fades after all the excited phosphor electrons have returned to their ground energy level. The frequency (or color) of the light emitted by the phosphor is proportional to the energy difference between the excited quantum state and the ground state.

RASTER-SCAN DISPLAYS :-

The most common type of graphics monitor employing a CRT is the raster-scan display, based on television technology.

In a raster-scan system, the electron beam is swept across the screen, one row at a time from top to bottom. As the electron beam moves across each row, the beam intensity is turned on and off to create a pattern of illuminated spots.



- Picture definition is stored in a memory area called the refresh buffer or frame buffer.

This memory area holds the set of intensity values for all the screen points.

- Stored intensity values are then retrieved from the refresh buffer and "painted" on the screen one row (scan line) at a time (in above Figure**).**
- Each screen point is referred to as a pixel or pel (shortened forms of picture element).
- Intensity range for pixel positions depends on the capability of the raster system. In a simple black-and-white system, each screen point is either on or off, so only one bit per pixel is needed to control the intensity of screen positions.
- For a bi-level system, a bit value of 1 indicates that the electron beam is to be turn on at that position,
- And a value of 0 indicates that the beam intensity is to be off.

- Additional bits are needed when color and intensity variations can be displayed. Up to 24 bits per pixel are included in high-quality systems, which can require several megabytes of storage for the frame buffer, depending on the resolution of the system.

A system with 24 bits per pixel and a screen resolution of 1024 by 1024 requires 3 megabytes of storage for the frame buffer. On a black-and-white system with one bit per pixel, the frame buffer is commonly called a bitmap.

For systems with multiple bits per pixel, the frame buffer is often referred to as a pixmap.

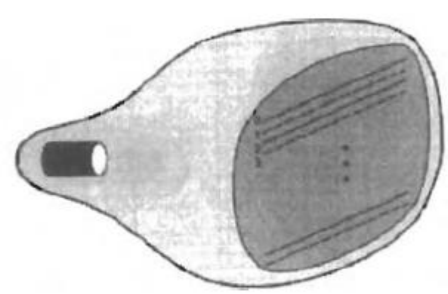
Refreshing on raster-scan displays is carried out at the rate of 60 to 80 frames per second, although some systems are designed for higher refresh rates. Sometimes, refresh rates are described in units of cycles per second, or Hertz (Hz), where a cycle corresponds to one frame. Using these units, we would describe

RANDOM-SCAN DISPLAYS

A random-scan display unit, a CRT has the electron beam directed only to the parts of the screen where a picture is to be drawn.

Random scan monitors draw a picture one line at a time and for this reason are also referred to as vector displays *(or* stroke-writing or calligraphic displays).

The Component lines of a picture can be drawn and refreshed by a random-scan system in any specified order which is shows in below figure.



(Fig: Interlacing scan lines on the raster scan system)

Refresh rate on a random-scan system depends on the number of lines to be displayed. Picture definition is now stored as a set of line drawing commands in an area of memory r e f e d to as the refresh display file.

Sometimes the refresh display file is called the display list, display program, or simply the refresh buffer.

To display a specified picture, the system cycles through the set of commands

After all line drawing commands have been processed, the system cycles back to the first line command in the list. Random-scan displays are designed to draw all the component lines of a picture 30 to 60 times each second.

High quality vector systems are capable of handling approximately 100,000 "short" lines at this refresh rate. When a small set of lines is to be displayed, each refresh cycle is delayed to avoid refresh rates greater than 60 frames per second. Otherwise, faster refreshing of the set of lines could bum out the phosphor.

Random-scan systems are designed for line drawing applications and cannot display realistic shaded scenes. Since picture definition is stored as a set of line drawing instructions and not as a set of intensity values for all screen points, vector displays generally have higher resolution than raster systems. Also, vector displays produce smooth line drawings because the CRT beam directly follows the line path.

COLOR CRT MONITORS:-

Color Monitors:

A color CRT monitor displays color picture by using a combination of phosphors that emit different colored light. By combining the emitted light a range of colors can be generated. Two basic methods for producing color displays are:

- Beam Penetration Method
- Shadow-Mask Method

Beam Penetration Method

Random scan monitors use the beam penetration method for displaying color picture. In this, the inside of CRT screen is coated two layers of phosphor namely red and green. A beam of slow electrons excites only the outer red layer, while a beam of fast electrons penetrates red layer and excites the inner green layer. At intermediate beam speeds, combinations of red and green light are emitted to show two additional colors- orange and yellow.

Advantages

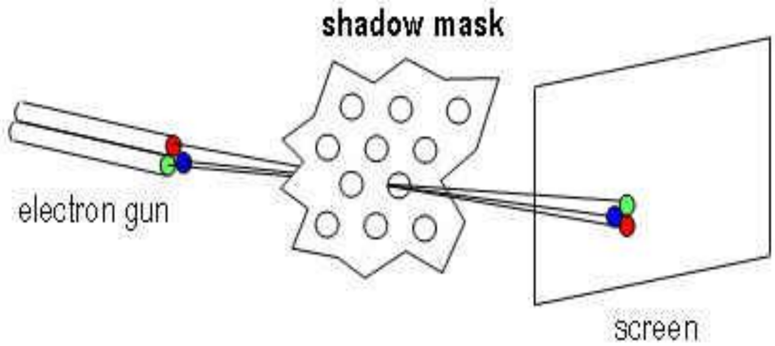
- Less expensive

Disadvantages

- Qualities of images are not good as comparatable with other methods
- Four colors are allowed only

Shadow Mask Method

Raster scan system are use shadow mask methods to produce a much more range of colors than beam penetration method. In this, CRT has three phosphor color dots. One phosphor dot emits a red light, second emits a green light and third emits a blue light. This type of CRT has three electrons guns and a shadow mask grid as shown in figure below:



In this figure, three electrons beams are deflected and focused as a group onto the shadow mask which contains a series of holes. When three beams pass through a hole in shadow mask they activate dot triangle as shown in figure below:

The colors we can see depend on the amount of excitation of red, green and blue phosphor. A white area is a reasult of all three dots with equal intensity while yellow is produced with green and red dots and so on.

Advantages

- Produce realistic images also produced different colors and shadows scenes.

Disadvantages

- low resolution
- expensive
- electron beam directed to whole screen

Color CRTs in graphics systems are designed as RGB monitors. These monitors use shadow mask method and take the intensity level for each gun. A RGB color system with 34 bits of storage per pixel is known as full color system or true color system.

DIRECT VIEW STORAGE TUBES:

- An Alternative method for maintaining a screen image is to store the picture information inside the CRT instead of refreshing the screen.

- DIRECT VIEW STORAGE TUBES stores the picture information as the phosphor-coated screen
- Two electron guns are used in DVST

1. Primary Gun-store the picture pattern
2. Flood Gun-Maintains the picture display.

ADVANTAGES:
Very complex pictures can be displayed at very high resolutions without ficker.

DISADVANTAGES:

- In DVST system ordinarily do not display color and that selected parts of a picture cannot be erased To eliminate a picture section the entire screen must be erased and the modified picture redrawn.
- The erasing and redrawing process can take several seconds for a complex picture.

FLAT-PANEL DISPLAYS:

Although most graphics monitors are still constructed with CRT .Other technologies are emerging that soon replace CRT monitors.

- Flat panel display is refered to class of video devices that have reduce volume weight and power requirements compared to a CRT.
- Flat panel display is that they are thinner than CRTs.
- Since we can write on some flat panel display they will soon be available as pocket notepads.

CURRENT USES:

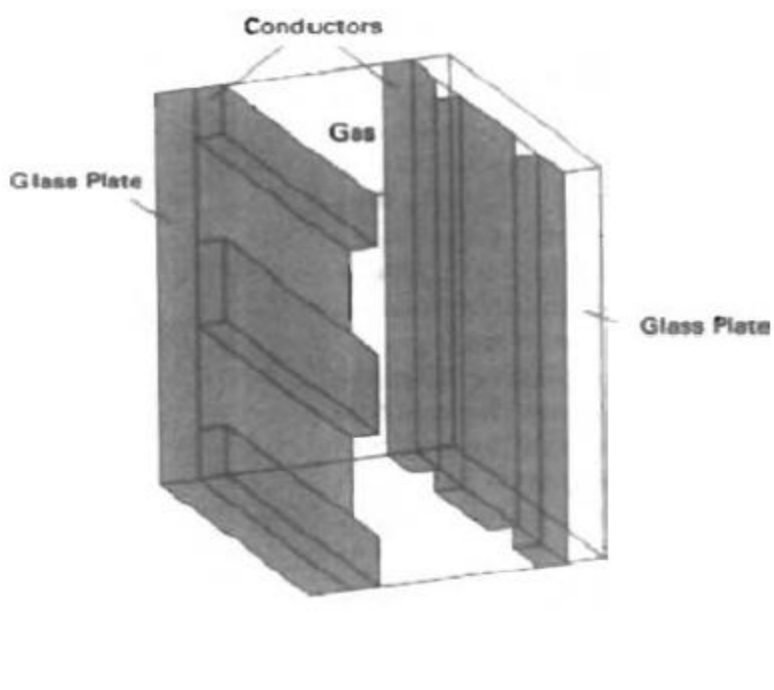
Small TV monitors, Calculators, Pocket video games, laptop computers and viewing of movies on airlines, advertisement boards in elevators and portable monitors etc.

Flat-panel displays can be divided into two types:

1. Emissive display

2. Non emissive display

- Emissive display- are device that convert electrical energy into light.
  Examples:-plasma panels,thin flim,electroluminescent display and light emitting diodes
- Non Emissive display-use optical effects to convert sunlight or light from other sources into graphics pattern
  Examples:liquid crystal device
- Plasma panel- also called gas-discharge displays are constructed by filling the region between two glass plates with a mixture of gasses
- Plasma display panels are most often seen as large flat televisions, while vacuum fluorescent displays are used in applications where the information content is fairly low, such as the displays on appliances or in automobiles. Field-emission displays are the most recent of these flat-panel technologies

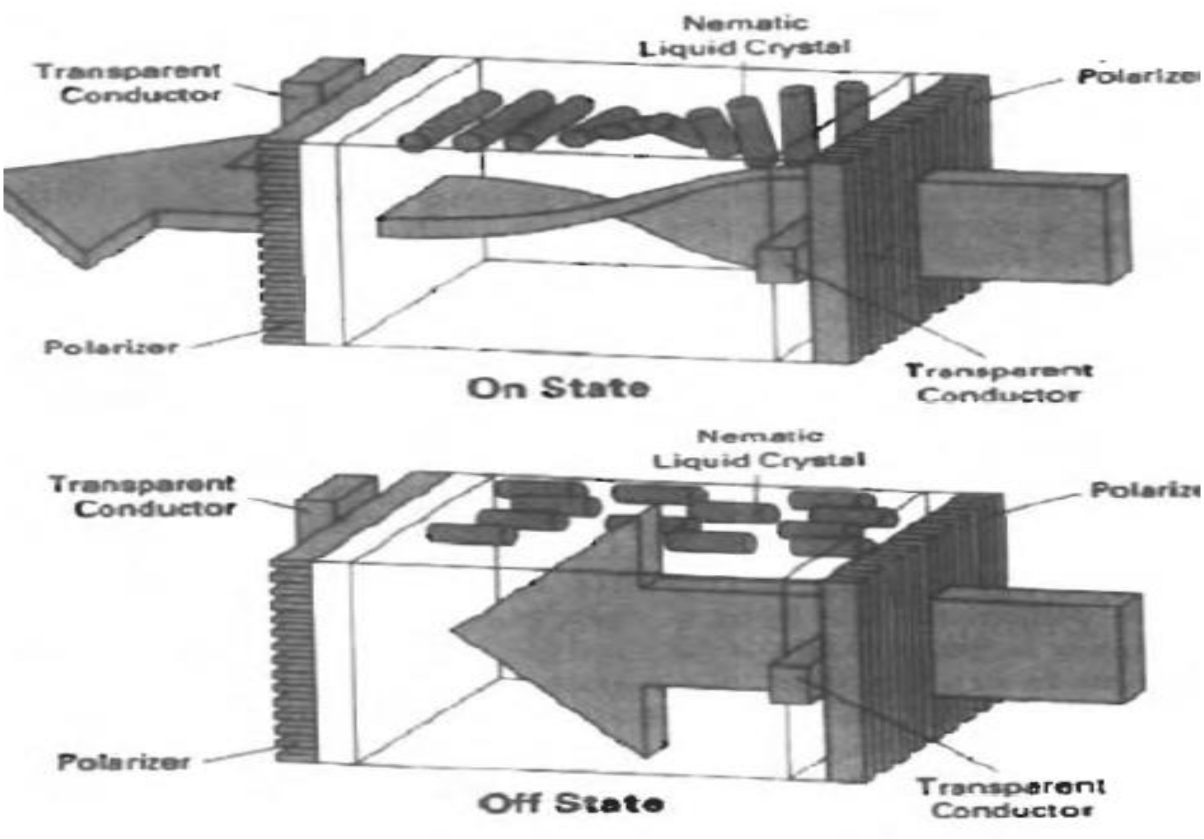


- Thin-Film electroluminescent displays:-are similar constructions to a plasma panel. The difference is that region between glass plates is filled with a phosphor .

- A third type of emissive device is the LIGHT -EMITTING DIODE (LED)-A matrix of diodes is arranged to form the pixel positions.

LIQUID CRYSTAL DISPLAY:-

- Liquid crystal displays (LCDS) are commonly used in small systems, such as calculators and portable, laptop computers.
- These non emissive devices produce a picture by passing polarized light from the surroundings or from an internal light **s o w** through a liquid-crystal material that can be aligned to either block or transmit the light.
- A flat-panel display can then be constructed with a nematic liquid crystal



- Two glass plates, each containing a light polarizer at right angles to the-other plate, sandwich the liquid-crystal material. Rows of horizontal transparent conductors are built into one glass plate, and columns of vertical conductors are put into the other plate.

- The intersection of two conductors defines a pixel position. Normally, the molecules are aligned as shown in the above figure "on state".

RASTER SCAN SYSTEMS:-

Raster graphics systems typically employ several processing units. In addition to the central processing unit, or CPU, a special-purpose processor, called the video controller or display controller, is used to control the operation of the display device.

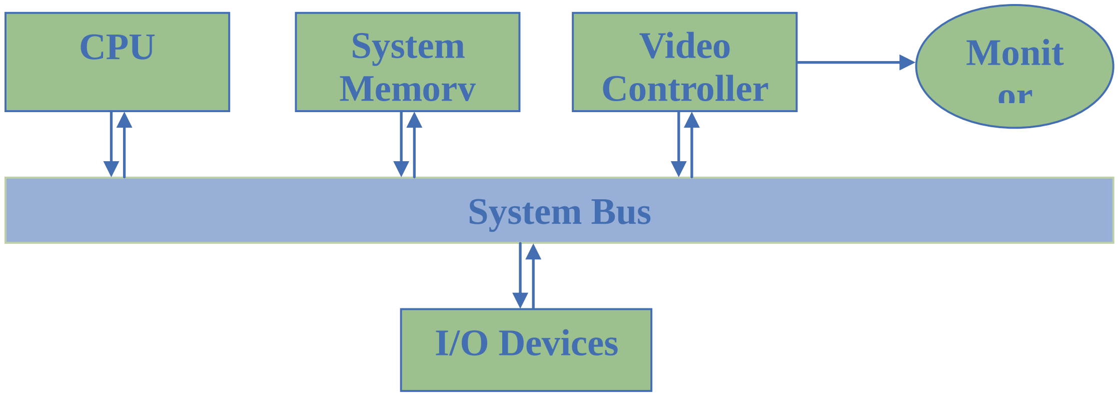
1. **Video Controller**
2. Raster scan display processor

Video Controller

A fixed area of the system memory is reserved for the frame buffer, and the video controller is given direct access to the frame-buffer memory.

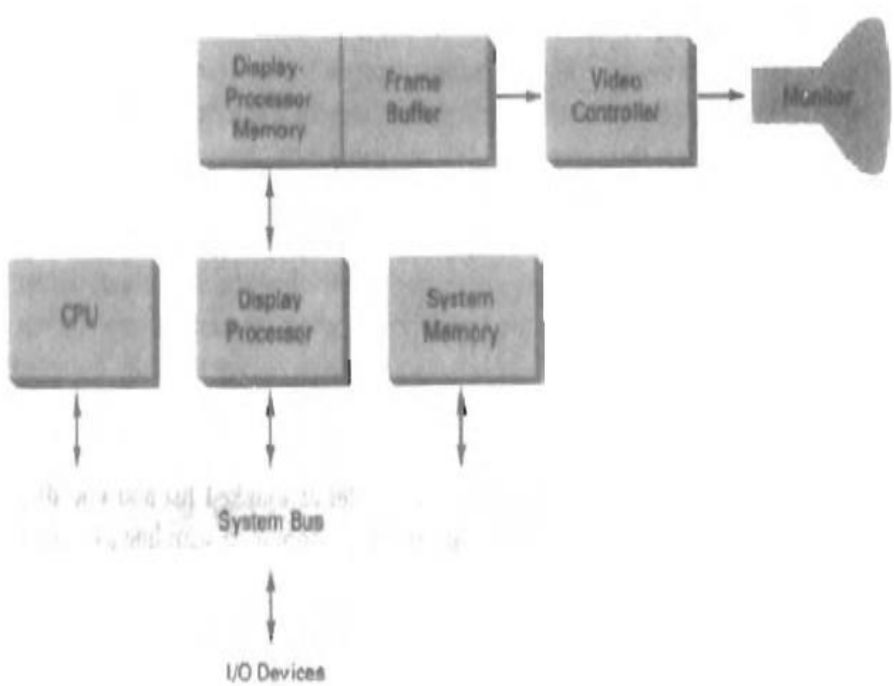
Frame-buffer locations, and the corresponding screen positions, are referenced in Cartesian coordinates.

The screen surface is then represented as the first quadrant of a two-dimensional system, with positive *x* values increasing to the right and positive y values increasing from bottom to top. Scan lines *are* then labeled from y, at the top of the screen to 0 at the bottom. Along each scan line, screen pixel positions are labeled from 0 to x_{max}



RASTER SCAN DISPLAY PROCESSOR:

The purpose of the display processor is to free the CPU from the graphics chores. In addition to the system memory, a separate display processor memory area can also be provided.



Architecture of raster scan display processor

RANDOM SCAN SYSTEM:

The organization of a simple random-scan (vector) system is shown in below figure. An application program is input and stored in the system memory along with a graphics package. Graphics commands in the application program are translated.

- The graphics package into a display file stored in the system memory.
- This display file is then accessed by the display processor to refresh the screen.
- The display processor cycles through each command in the display file program once during every refresh cycle. Sometimes the display processor in a random-scan system is referred to as a display processing unit or a graphics controller.

3D-VIEWING DEVICES:-

Graphics monitors for the display of three-dimensional scenes have been devised using a technique that reflects a CRT image from a vibrating, flexible mirror.

The operation of such a system is demonstrated in below figure. As the varifocal mirror vibrates, it changes focal length. These vibrations are synchronized with the display of an object on a CRT so that each point on the object is reflected from the mirror into a spatial position corresponding to the distance of that point from a specified viewing position. This allows us to walk around an object or scene and view it from different side.

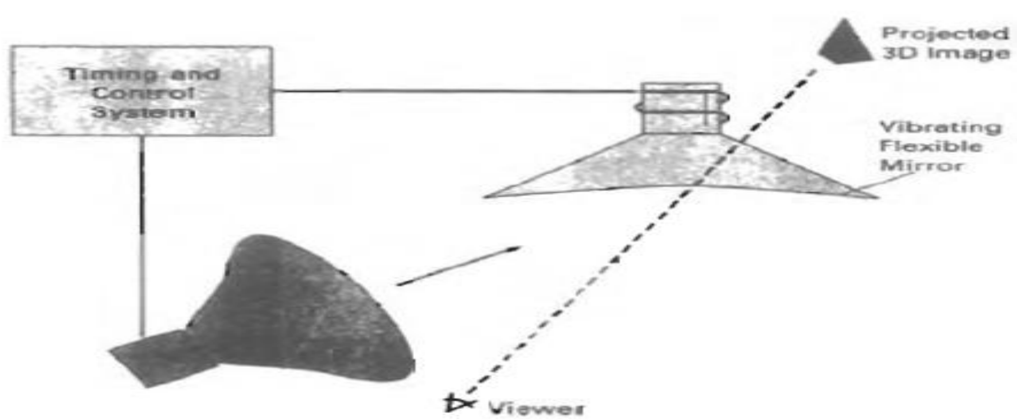


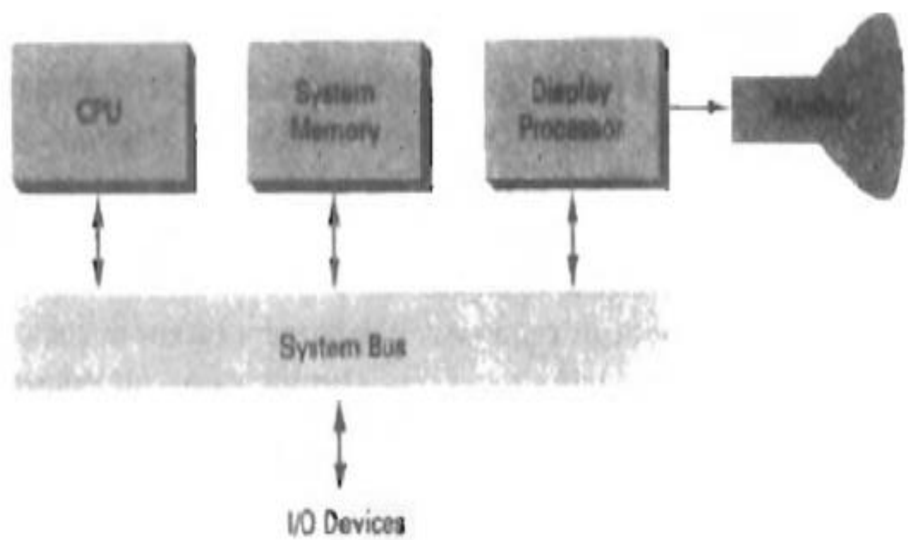
Fig: Operation of a 3D Display system.

RANDOM SCAN SYSTEM:-

The organization of a simple random-scan (vector) system is shown in below figure. An application program is input and stored in the system memory along with a graphics package.

Graphics commands in the application program are translated by the graphics package into a display file stored in the system memory. This display file is then accessed by the display processor to refresh the screen.

The display processor cycles through each command in the display file program once during every refresh cycle. Sometimes the display processor in a random-scan system is referred to as a display processing unit or a graphics controller.



Graphics patterns are drawn on a random-scan system by directing the electron beam along the component lines of the picture.

Lines are defined by the values for their coordinate endpoints, and these input coordinate values are converted to x and y deflection voltages. A scene is then drawn one line at a time by positioning the beam to fill in the line between specified endpoints.

POSSIBLE QUESTIONS

UNIT I

Part-A

Online Examinations (One marks)

1. Bresenham's circle generating algorithm will take reflections of ________

   a) Two octets b) **One octet** c) Three octets d)Four octets

2. Eight-way symmetry is used by reflecting each calculated point around each ___ axis

   a) 35° b)180° c)**45°** d)90°

3. A _______ is a small raster containing the relative locations of the pixels that are used to represent the character

   a) Frame buffer b) **Mask** c) Display d) Shadow

4. In Line display of characters ________ test is used to determine the intersecting lines

   a) Integration b) Black box c) Unit d)**Minmax**

5. When ____________ is used for a picket fence problem, the distance between pickets are kept close to their true distance

   a) Private aliasing b) **local aliasing** c) Global aliasing d) Public aliasing

6. The view plane is also called __________

   a) **Projection plan** b) view direction c) mapping function d)reference vector

7. The Center of projection is called the ____

   a) **View point** b) view volume c) viewport d) reference vector

8. The perception of color arises from __________ entering our visual system.

   a) Image b) view volume c) **light** d) color

9. Light is________ energy.

   a) Electrical b) mechanical c) kinetic d) **electromagnetic**

10. The wavelength of the visual light ranges form _______nanometer.

   a) 100 to 500 b) **400 to 700** c) 30 to 90 d)600 to 1400

Part B

Essay Type (8 Marks)

1. Define Computer Graphics. List out the applications used in computer graphics.
2. With neat diagram explain about the refresh CRT?
3. What are the applications of computer graphics?
4. Define Random scan/Raster scan displays.
5. What is meant by refreshing of the screen?
6. List out the merits and demerits of DVST?
7. What do you mean by GUI?Explain in detail
8. Discuss about pixels and frame buffer in detail
9. Explain briefly about direct view storage tubes.
10. Give a detailed account on flat panel displays.
11. Explain in detail about Refresh Cathode Ray Tube.
12. Discuss about Color CRT Monitors.
13. Give a detailed account on Raster and Random Scan systems

KARPAGAM ACADEMY OF HIGHER EDUCATION

COIMBATORE - 21

DEPARTMENT OF COMPUTER SCIENCE

CLASS : III B.Sc COMPUTER SCIENCE

BATCH : 2015-2018

Part -A Online Examinations **(1 mark questions)**

SUBJECT: COMPUTER GRAPHICS **SUBJECT CODE: 15CSU503**

	UNIT I					
S.NO	**Questions**	**option1**	**option2**	**option3**	**option4**	**Answer**
1	________ is the discipline of producing picture or images using a computer which include modeling, creation, etc.	HTML	Computer Graphics	Programming	Scipting Language	Computer Graphics
2	CRT stands for	Control Ray tube	Cathode reverse tube	Cathode Ray Tube	Buffer	Cathode Ray Tube
3	Cathode ray is called as	Fluorescent	Electron beam	Electromagnetic	Phosphor	Electron beam
4	___________ inside the cylinderical cathode structure.	Control Grid	Pixel	Filament	CRT	Filament
5	Electrostatic focusing is commonly used in	television and computer grapics monitor	Programming Language	Spectrometer	Focusing System	television and computer grapics monitor
6	The ___________ display is used in television sets.	Persistance	Pixel	Phosphor	Cathode Ray Tube	Cathode Ray Tube
7	Duration of phospherscene exhibited by the phosphor is called	Pixel	Persistance	Electromagnetic	CRT	Persistance
8	Picture definition is stored in a memory area is called	Memory	Picture definition	Frame Buffer	Fluorecent	Frame Buffer
9	The ___________ in CRT is needed to force the electron beam to converge into a small spot as it strikes the phosphor.	Focusing system	Control grid	Accessing point	Pixel Position	Focusing system

10	Deflection of electron beam can be controlled either with	Potensiometer	Magenetic diode	Pixel or Pel	Electricfield or magnetic field.	Electricfield or magnetic field.
11	The maximum number of points that can be displayed without overlapping on a CRT is referred as the ______	Picture definition	Resolution	Magnetic field	None of the Above	Resolution
12	―――――― is defined as the time it takes the emitted light from the screen of decay to one- tenth of its original intensity.	Raster scan display	Persistance	Random Graphics System	Random Scan device	Persistance
13	The most common type of graphics monitor employing a CRT Is the	Raster scan display	Random Scan device	Picture Definition	Pixel Position	Raster scan display
14	In a raster scan system the electron beam is swept across the	Raster scan display	Random Graphics System	Picture Definition	phosphor screen	phosphor screen
15	In raster scan method electron beam passes	Left to right	Top to bottom	bottom to left	Right to left	Top to bottom
16	Picture definition is stored in a memory area called	Raster scan display	Random Graphics System	Picture Definition	Referesh buffer or Frame buffer	Referesh buffer or Frame buffer
17	Each rows in raster scan display are called as	Scan lines	raster line	Random line	Pel	Scan lines
18	Picture stored area is called as	Emissive displays	Flat panel displa	Pixel or Pel	optical effects	Pixel or Pel
19	On black and white system with one bit per pixel the frame buffer is commonly called a	Mega byte	Byte Stored	Bitmap and Pixmap	None of the above	Bitmap and Pixmap
20	Refreshing on raster scan displays is carried out at the rate of	60to80 per sec	20 to 40 per sec	20 to 60 per sec	80 to 100 per sec	60to80 per sec
21	―――――― refers to a class of video devices that have reduced volume,weight and power requirements	Emissive displays	Flat panel displa	liquid crystal	optical effects	Flat panel display
22	Flat panel display is than CRT	Lighter	Heavier	Sharper	Thinner	Thinner
23	Flat panel display is divided into categories	two	three	four	five	two
24	――――― convert electrical energy into light	Flat panel display	liquid crystal	optical effects	Emissive displays	Emissive displays
25	Flat CRT's in which electron beams are accelarated parallel to the screen,then deflected degree to the screen.	40˚	90˚	80˚	120˚	90˚

26	Nonemissive displays use effects to convert sunlight.	optical effects	Refresh buffer	liquid crystal	Gas-discharge displays	optical effects
27	Nonemissive flat panel display is a device.	Flat panel display	liquid crystal	optical effects	Emissive	liquid crystal
28	Plasma panel also called	Gas-discharge displays	Thin film electro	Video controller or display controller	Virtual reality	Gas-discharge displays
29	Picture definition is stored in	Flat panel display	Refresh buffer	Thin film electroluminescent	Video controller or display controller	Refresh buffer
30	Firing voltage are applied to refresh the pixel position times per second	20	60	40	69	60
31	____________ are similar in construction to a plasma panel	display list,display program or refresh buffer	Thin film electro	scan conversion	display processing unit or graphics controller	Thin film electroluminescent
32	__________ is used to control the operation of the display device	Raster scan	CRT	RCRT	Video controller or display controller	Video controller or display controller
33	Frame-buffer locations,and the corresponding screen position,are referenced in	Polynomial Coordination	Beam penetration	Cartesian coordinates	Rastor coordination	Cartesian coordinates
34	Scan lines are then labeled from Ymax at the top of the screen to at the bottom	0	1	3	6	0
35	Raster System sometimes referred to as	scan conversion	graphics controll	Polynomial Coordination	Beam penetration	graphics controller or display co-processor
36	The digitization process is called	scan conversion	Raster scan	CRT	RCRT	scan conversion

37	Random-scan system is refered to as a	Pixel beam	scan conversion	Phosphor	Control grid	display processing unit or graphics controller
38	____________ are drawn on a random scan system by directly the electron beam along the component lines of the picture	Data View Storage	Direct Viewing Device	display processing unit or graphics controller	Raster scan	Graphics patterns
39	The input co-ordinate values are converted to deflection voltages	x,y,z	y and z	x and z	X and Y	X and Y
40	Refresh display file is called the	Phosphor	Control grid	display list,display program or refresh buffer	Electromagnetic	display list,display program or refresh buffer
41	Random-scan displays are designed to draw all the component lines of picture in times for each second ______	30to60	50to 60	10 to 40	0 to 50	30to60
42	Refresh cycle is displayed to avoid refresh raster greater than per second	20frames	60 frames	10frames	100frames	60 frames
43	A CRT monitor displays color pictures by using combination of	potassium	Refresh buffer	Electromagnetic	phosphors	phosphors
44	Beam penetration method for displaying color pictures has been used with	Random-scan monitors	Raster scan	CRT	RCRT	Random-scan monitors
45	A beam of slow electrons excites only the outer layer	Red	Green	Blue	Yellow	Red
46	A beam of fast electron excite the inner layer	Red	Green	Blue	Yellow	Green
47	Combinations of red and green light are emitted to show two additional colors	Red and Green	Blue and Orange	orange and yellow	RGB	orange and yellow
48	Shadow-mask methods are commonly used in	Random-scan monitors	Raster scan system	CRT	RCRT	Raster scan system

49	A shadow-mask CRT has three phosphor color dots at each position	Random	pixel	Raster	phosphors	pixel
50	RGB refers to	Red-Green-Blue	Color CRT	Random Graphics System	Black	Red-Green-Blue
51	A gray-scale image is typically coded with bits per pixel	2	4	9	8	8
52	TIFF refers to	Tagged Image Focus Format	Tagged Image F	TIFF Image Format	Trasfer Image File Format	Tagged Image File Format
53	Colors CRT in graphics systems are designed as	RGB monitors	Pixel	Phosph	CMYK	RGB monitors
54	High-quality raster graphics systems have per pixel	12bits	64bits	24bits	128bits	24bits
55	RGB color system with storage per pixel	12bits	64bits	24bits	128bits	24bits
56	DVST refers to	Direct View Storage Tubes	Direct Visible Storage Tubes	Data View Storage	Direct Viewing Device	Direct View Storage Tubes
57	________ electron guns are used in DVST	Phosphor	Control grid	Pixel or Pel	primary gun and flood gun	primary gun and flood gun
58	________ is stores the picture pattern	Phosphor	Primary gun	Pixel or Pel	Direct Visible Storage Tubes	Primary gun
59	________ is maintains the picture display	Flood gun	Pixel or Pel	Phosphor	primary gun	Flood gun

KARPAGAM ACADEMY OF HIGHER EDUCATION

(Deemed University Established Under Section 3 of UGC Act 1956)

Coimbatore - 641021.

(For the candidates admitted from 2015 onwards)

DEPARTMENT OF COMPUTER SCIENCE, CA & IT

SUBJECT : COMPUTER GRAPHICS
SEMESTER : V
SUBJECT CODE: 15CSU503 CLASS : III B.Sc.CS

UNIT-II

Input Devices: Keyboards-Mouse –Track Ball and Space ball-Joysticks-Data Glove- digitizers-Image Scanners- Touch Panels-Light Pens-Voice Systems-**Hard Copy Devices**: Printers and Plotters

TEXT BOOK

1. Donald Hearn and M. Pauline Baker. 2007. *Computer Graphics-C Version*, 2nd Edition, Pearson Education, New Delhi.
   (Page Nos. : 24-54, 56-77, 80-92, 103-118, 204-215, 236-256, 427-443, 458-463, 490-495, 502-505)

REFERENCES

1. Amarendra N. Sinha. 2008. *Computer Graphics*, 1st Edition ,Tata McGraw Hill, New Delhi.
2. Foley, Vandam, Feiner and Hughes. 1999. *Computer Graphics Principles and Practices*, 2nd Edition ,Addison Wesley, Singapore.
3. Evangeline D., Anita, S. 2016. *Computer Graphics And Multimedia Insights, Mathematical Models And Programming Paradigms*. PHI Learning Pvt. Ltd.
4. Rae Earnshaw.2014.*Computer Graphics: Developments in Virtual Environments*, 2nd Edition ,academic Press, New Delhi.

WEB SITES:

www.cgshelf.com
www.cg**tutorials**.com
www.allgraphicdesign.com

UNIT II
INPUT DEVICES

INPUT DEVICES:-

Various devices are available for data input on graphics workstations. Most systems have a keyboard and one or more additional devices specially designed for iterative input.

These include a mouse, trackball, space ball, joystick, digitizers, dials, and button boxes. Some other inputs devices are used In particular applications are data gloves, touch panels, image scanners, and voice systems.

KEYBOARD:-

An alphanumeric keyboard on a graphics system is used primarily as a device for entering text strings.

- The keyboard is an efficient device for inputting such non graphic data as picture labels associated with a graphics display.
- Keyboards can also be provided with features to facilitate entry of screen coordinates, menu selections, or graphics functions.
- Cursor-control keys and function keys are common features on general purpose keyboards.
- Function keys allow users to enter frequently used operations in a single keystroke, and cursor-control keys can be used to select displayed objects or coordinate positions by positioning the screen cursor.

Other types of cursor-positioning devices, such as a trackball or joystick, are included on some keyboards. Additionally, a numeric keypad is, often included on the keyboard for fast entry of numeric data. Typical examples of general-purpose keyboards.

Alphanumeric Keys - letters and numbers.

Punctuation Keys - comma, period, semicolon, and so on.

Special Keys - function keys, control keys, arrow keys, Caps Lock key, and so on.

MOUSE:

A mouse is small hand-held box used to position the screen cursor. Wheels or rollers on the bottom of the mouse can be used to record the amount and direction of movement.

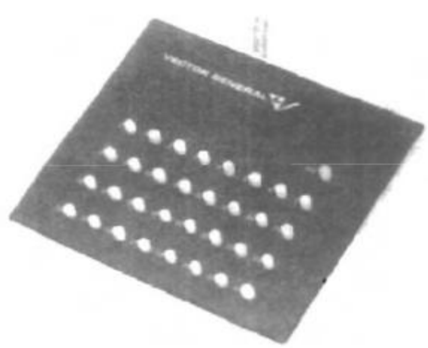
Another method for detecting mouse motion is with an optical sensor. For these systems, the mouse is moved over a special mouse pad that has a grid of horizontal and vertical lines. The optical sensor detects movement across the lines in the grid.

A mouse can be picked up and put down at another position without change in cursor movement; it is used for making relative change in the position of the screen cursor. One, two, or three button usually included on the top of the mouse for signaling the execution of some operation, such as recording cursor position or invoking a function.

MOUSE

Additional devices can be included in the basic mouse design to increase the number of allowable input parameters. The Z mouse includes three buttons, a thumbwheel on the side, a trackball on the top, and a standard mouse ball underneath.

This design provides six degrees of freedom to select Input Devices spatial positions, rotations, and other parameters. With the Z mouse, we can pick up an object, rotate it, and move it in any direction, or we can navigate our viewing position and orientation through a three-dimensional scene. Applications of the Z mouse include virtual reality, CAD, and animation.



TRACKBALL AND SPACE BALL:-

Trackball is a ball that can be rotated with the fingers or palm of the hand, to produce screen-cursor movement.

Potentiometers, attached to the ball, measure the amount and direction of rotation. Trackballs are often mounted on keyboards or other devices such as the Z mouse.

- While a trackball is a two-dimensional positioning device,

SPACE BALL:-

Space ball provides six degrees of freedom. Unlike the trackball, a space ball does not actually move. Strain gauges measure the amount of pressure applied to the space ball to provide input for spatial positioning and orientation as the ball is pushed or pulled in various directions.

- Space balls are used for three-dimensional positioning and selection operations in virtual-reality systems, modeling, animation, CAD, and other applications.

JOYSTICKS:-

A joystick consists of a small, vertical lever (called the stick) mounted on a base that is used to steer the screen cursor around.

Most joysticks select screen positions with actual stick movement others respond to pressure on the stick. Some joysticks are mounted on a keyboard others function as stand-alone units.

The distance that the stick is moved in any direction from its center position corresponds to screen-cursor movement in that direction. Potentiometers mounted at the base of the joystick measure the amount of movement, and springs return the stick to the center position when it is released. One or more buttons can be programmed to act as input switches to signal certain actions once a screen position has been selected.

In another type of movable joystick, the stick is used to activate switches that cause the screen cursor to move at a constant rate in the direction selected. Eight switches, arranged in a circle, are sometimes provided, so that the stick can select any one of eight directions for cursor movement. Pressure sensitive joysticks, also called isometric joysticks, have a non movable stick.

JOYSTICKS

DATA GLOVE:-

Data glove that can be used to grasp a "virtual" object. The glove is constructed with a series of sensors that detect hand and finger motions.

- Electromagnetic coupling between transmitting antennas and receiving antennas is used to provide information about the position and orientation of the hand.

- The transmitting and receiving antennas can each be structured as a set of three mutually perpendicular coils, forming a three-dimensional Cartesian coordinate system.

- Input from the glove can be used to position or manipulate objects in a virtual scene.

- A two-dimensional projection of the scene can be viewed on a video monitor, or a three-dimensional projection can be viewed with a headset.

A virtual reality screen displayed on a 2-D video monitor with input from data glove and a space ball

DIGITIZERS:-

A common device for drawing, painting, or interactively selecting coordinate positions on an object is a digitizer. These devices can be used to input coordinate values in either a two-dimensional or a three-dimensional space.

Typically, a digitizer is used to scan over a drawing or object and to input a set of discrete coordinate positions, which can be joined with straight- line segments to approximate the curve or surface shapes.

One type of digitizer *is* the graphics tablet (also referred to as a data tablet), which is used to input two-dimensional coordinates by activating a hand cursor or stylus at selected positions on a flat surface. A hand cursor contains cross hairs for sighting positions, while a stylus is a pencil-shaped device that is pointed at positions on the tablet.

- Many graphics tablets are constructed with a rectangular grid of wires embedded in the tablet surface.
- Three-dimensional digitizers use sonic or electromagnetic transmissions to word positions. One electromagnetic transmission method is similar to that used in the data glove.

- A coupling between the transmitter and receiver is used to compute the location of a stylus as it moves over the surface of an object.

DIGITIZERS

IMAGE SCANNERS:-

Drawings, graphs, color and black-and-white photos, or text can be stored for computer processing with an image scanner by passing an optical scanning mechanism over the information to be stored.

The gradations of gray scale or color are then recorded and stored in an array. Once we have the internal representation of a picture, we can apply transformations to rotate, scale, or crop the picture to a particular screen area. We can also apply various image-processing methods to modify the array representation of the picture.

For scanned text input, various editing operations can be performed on the stored documents. Some scanners are able to scan either graphical representations or text, and they come in a variety of sizes and capabilities.

IMAGE SCANNERS

TOUCH PANELS:-

Touch panels allow displayed objects or screen positions to be selected with the touch of a finger. A typical application of touch panels is for the selection of processing options that are represented with graphical icons. Some systems, such as the plasma panels are designed with Touch screens

Other systems can be adapted for touch input by fitting a transparent device with a touch sensing mechanism over the video monitor screen. Touch input can be recorded using optical, electrical, or acoustical methods. Optical touch panels employ a line of infrared light-emitting diodes (LEDs)

Along one vertical edge and along one horizontal edge of the frame. The opposite vertical and horizontal edges contain light detectors. These detectors are used to record which beams are interrupted when the panel is touched.

The two crossing beams that are interrupted identify the horizontal and vertical coordinates of the screen position selected. Positions tin be selected with an accuracy of about ¼ inch With closely spaced LEDs,

The LEDs operate at infrared frequencies, so that the light is not visible to a user. The arrangement of LEDs in an optical touch panel that is designed to match the color and contours of the system to which it is to be fitted.

An electrical touch panel is constructed with two transparent plates separated by a small distance

1. **One of the plates is coated with a conducting material,**
2. **and the other plate is coated with a resistive material**

In acoustical touch panels, high-frequency sound waves are generated in the horizontal and vertical directions across a glass plate. Touching the screen causes part of each wave to be reflected from the finger to the emitters.

The screen position at the point of contact is calculated from a measurement of the time interval between the transmission of each wave and its reflection to the emitter.

TOUCH PANEL

LIGHT PENS:-

In the below figure shows the design of one type of light pen. Such pencil-shaped devices are used to select screen positions by detecting the light coming from points on the CRT screen They are sensitive to *the* short burst of light emitted from the Phosphor coating at the instant the electron beam strikes a particular point. Other Light sources, such as the background light in the room, are usually not detected by a light pen.

An activated light pen, pointed at a spot on the screen as the electron beam lights up that spot, generates an electrical pulse that causes the coordinate position of the electron beam to be recorded. As with cursor-positioning devices, recorded Light-pen coordinates can be used to position an object or to select a processing option.

Light pen

- Although Light pens are still with us, they are not as popular.

DISADVANTAGES:-

- Command to other input devices that have been developed. For one, when a light pen is pointed at the screen, part of the screen image is obscured by the hand and pen.
- Light pens require special implementations for some applications because they cannot detect positions within black areas.
- To be able to select positions in any screen area with a light pen, we must have some nonzero intensity assigned to each screen pixel. In addition, light pens. Sometimes give false readings due to background lighting in a room.

VOICE SYSTEMS:-

Speech recognizers are used in some graphics workstations as input devices to accept voice commands the voice-system input can be used to initiate graphics operations or to enter data.

These systems operate by matching an input against predefined dictionary of words and phrases. A dictionary is set up for a particular operator by having, the operator speak the command words to be used into the system. Each word is spoken Several times, and the system analyzes the word and establishes a frequency pattern for that word in the dictionary along with the corresponding function to be performed.

Later, when a voice command is given, the system searches the dictionary for a frequency-pattern match. Voice input is typically spoken into a microphone mounted on a headset, as in below

Fig. The microphone is designed to minimize input of other background sounds. If a different operator is to use the System, the dictionary must be reestablished with that operator's voice patterns. Voice systems have some advantage over other input devices, since the attention of the operator does not have to be switched from one device to another to enter a command.

A speech recognition system

HARD-COPY DEVICES:-

PRINTERS:

Printers are the most commonly used output device. They are used to print output on the paper. The output may be in the form of characters, symbols and graphics information printed on paper is called hardcopy

The various types of printers in used today are

1. **Dot-Matrix Printers**
2. **Inkjet Printers**
3. **Drum Printers**
4. **Laser Printers.**

Printers produce output by either

1. **Impact Printer**
2. **Non impact Printer**

1. Impact Printer

The types of printers that produce output on paper by striking the print hammer or wheel against an inked ribbon are called impact printers. Impact printers work like typewriter. They can print characters and graphics on the paper.

Impact printers are slower in printing and produce low quality output. The printing speed of these printers is measured in characters or lines per minute. They also produce more noise during printing. Today they are not commonly used.

The examples of impact printers are:

1- Dot matrix printer

2- Daisy Wheel printer

3- Line printer

DOT MATRIX PRINTERS:

A dot matrix printer is an impact character printer. It makes a hardcopy by printing one character at a time. Its printing speed is from 200 to 1000 or more characters per minute.

Dot matrix printer contains a print-head with a matrix of small pins arranged in rows and columns. Dot matrix printer produces output on paper by striking pins against an ink ribbon. Usually, a dot matrix printer uses 100 to 300 dots per inch (DPI) to print output on the paper. Print-heads are avialable with 9, 18 or 24 pins. The dot matrix printer with 24-pins provides best quality printout.

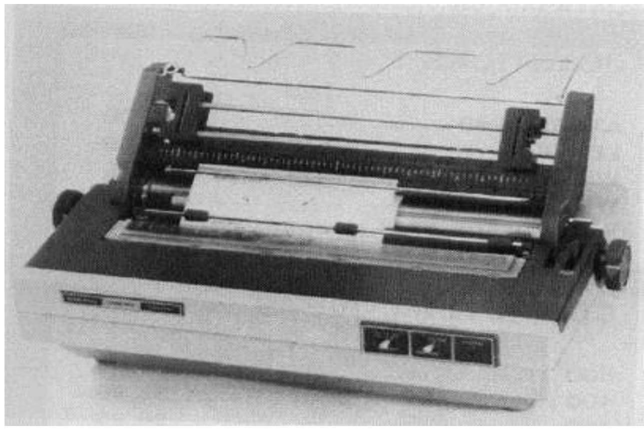
Dot matrix printers are used with personal computer. They are less expensive. The printout quality of these printers is not bad. They also produce more noise during printing.

DOT MATRIX PRINTER

DAISY WHEEL PRINTER:-

Daisy wheel printer is also an impact character printer. It is similar to typewriter. It has a print wheel with a series of petals. This wheel is known as daisy wheel. Each petal of daisy wheel contains a character at its end.

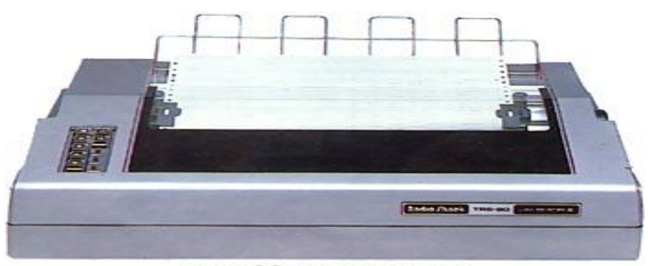
A motor rotates the wheel. When the desired character reaches at the print position on the paper, a hammer strikes a petal against the ribbon. In this way, a character is printed on the paper. This printer is slower than dot-matrix printer. However, its print quality is better than dot matrix printer



DAISY WHEEL PRINTER

LINE PRINTER:-

Line printer is an impact printer. it is very fast printer. It prints a complete line of characters at a time. The printing speed of line printer is measured in lines per minute (lpm). It is up to 3000 lines per minute.



Line printers are normally used with mainframe and mini computers.

Two types of line printers are:-

1. **Chain Printer**
2. **Band Printer**

2. NON IMPACT PRINTERS:-

The printers that produce output on paper without striking the paper are known as non-impact printers. They Use Electrostatic, inkjet, and thermal technologies for printing.

Non-Impact printers are faster and produce high quality output than impact printers. They can print up to 24 pages per minute. They produce no noise during printing. These printers are costly than impact printers

The Examples of Non-Impact printers are:

1- Laser Printer

2- Inkjet Printer

3- Thermal Printer

Laser Printer

Laser stands for Light Amplification by Simulated Emission of Radiation. A laser printer is the fastest and high quality non-impact printer. It works like a photocopier.

The laser printer transfers the image of output on paper using LASER technology and toner. Toner is an ink powder. It is used in laser printers and photocopiers also

Laser Printer

The laser printer has a special drum inside it. first the image of output is created on the drum, and then it is transferred from drum to paper. The image of output is created on the drum by throwing magnetic ink powder in the form of microscopic dots.

These dots can be from 300 dpi to 1200 dpi (dpi means dots per inch and these dots refers to microscopic dots).

The Laser printer can print both text and graphics in very high quality resolution. Laser printer prints one page at a time. The laser printers are, therefore also called page printers. The printing speed of laser printer is about 4 to 32 pages per minute for microcomputer and up to 200 pages per minute for mainframe computers.

Ink Jet Printer

Ink-jet printer is type of non-impact printer. It creates output on paper by spraying tiny drops of liquid ink. Inkjet printer has print-head that can spray very fine drops of ink. It consists of print cartridge filled with liquid ink and has small nozzles in form of matrix.

Like dot matrix printer, the combination of nozzles is activated to form the shaper of character or image on the paper by spraying the liquid ink. These printers have resolution ranging from 300 to 720 dpi.

INK JET PRINTER

The ink-jet printers have low price than laser printers. They are also slower and have low print quality than laser printer. However, they are faster and have high print quality than dot matrix printers. The printing speed of ink-jet printer is from 1 to 6 pages per minute.

Thermal Printer

Thermal printer is another type of non-impact printer. It can only print output on a special heat sensitive waxy paper. The image if the output is created on the waxy paper by burning dots on it. For colored output, colored waxy sheets are used.

THERMAL PRINTER

Thermal Printer produces a high quality printout. It is quite expansive as compared to other non-impact printers.

POSSIBLE QUESTIONS
Unit-II
Part-A

Online Examinations (One marks)

1. ________ is the discipline of producing picture or images using a computer which include modeling, creation, etc.
   a) HTML b) **Computer Graphics** c) Programming d)Scripting Language
2. CRT stands for ______
   a) Control Ray tube b) Cathode reverse tube c) **Cathode Ray Tube** d) Buffer
3. The _________ in CRT is needed to force the electron beam to converge into a small spot as it strikes the phosphor.
   a) **Focusing system** b)Control grid c)Accessing point d)Pixel Position
4. Picture stored area is called as _______
   a) Emissive displays b)Flat panel display c)**Pixel or Pel** d)optical effects
5. An alphanumeric keyboard on a graphic system is used primarily as a device for entering _______
   a) **Text string** b) Numeric c) Alphanumeric d) String
6. The _________key is used to co-ordinate position by positioning the screen cursor.
   a) **Cursor control key** b)Mouse c)Optical touch d)Function key
7. _________ is small hand-held box used to position the screen cursor.
   a) Keyboard b) Monitor c) **mouse** d) CPU
8. __________detects movement across the lines in the grid.
   a) Digital b) **optical sensor** c) Graphics d) Analog
9. _________ attached to the ball, measure the amount and direction of rotation.
   a) Stegnometer b) Graphics monitor c) phosphor d) **Potentiometer**
10. Some monitors use a technique called ___________ to double their refreshing rate.
    a) Flickering b) **interlacing** c) doubling d) persistence

Part B

Essay Type (8 Marks)

1. Define printer? List out the different types of printer.
2. What are the features of Ink jet Printers?
3. Explain in detail about Light pen.
4. Define Impact printer and its types?
5. Explain in detail about track ball and space ball
6. Write short notes on Touch panels.
7. Explain in detail about Image Scanners.
8. Explain in detail about data glove.
9. Explain in detail about Digitizers and Joysticks.
10. Describe Impact Printer and its types
11. Explain in detail about image scanners and mouse.
12. Illustrate about the hard copy device in details.
13. Explain in detail about any 4 input devices with its operations.
14. Discuss about the Character printer and Ink-jet printer in detail.
15. Explain in detail about image scanners and Voice Systems.

KARPAGAM ACADEMY OF HIGHER EDUCATION

COIMBATORE - 21

DEPARTMENT OF COMPUTER SCIENCE

CLASS : III B.Sc COMPUTER SCIENCE

BATCH : 2015-2018

Part -A Online Examinations **(1 mark questions)**

SUBJECT: COMPUTER GRAPHICS **SUBJECT CODE: 15CSU503**

UNIT II

S.no	Questions	option1	option2	option3	option4	Answer
1	An alphanumeric keyboard on a graphic system is used primarily as a device for entering	Text string	Numeric	Alphanumeric	String	Text string
2	________ and ________ keys are common features on general purpose in keyboards	Mouse	Touch screen	Cursor-control	Optical touch	Cursor-control key,function keys
3	The ________ key is used to co-ordinate position by positioning the screen cursor.	Cursor control key	Mouse	Optical touch	Function key	Cursor control key
4	Another method for detecting mouse motion is __	Primary	Control	Input	optical sensor	optical sensor
5	The tablet use sound waves to detect a system position	Graphics	Acqustic or sonic	Stegnometer	Graphics monitor	Acqustic or sonic
6	Three dimensional digitizers use sonic or ________ transmission to record positions	Electrical Touch panels	Acoustic touch panal	Phosphor	Electromagnetic	Electromagnetic
7	Which dimensional digitizer designed for transmission to apple macintosh computers	Three dimensional	keyboard	Space balls	two dimensional	Three dimensional
8	The plasma panels are designed with ________	Digital	optical sensor	touch screen	Electrical touch pannels	touch screen
9	Touch input can be recorded using ________	optical & electrical	Stegnometer	Graphics monitor	phosphor	optical & electrical
10	________ employe a line of infrared light emitting diodes	Trackballs	Acqustic or sonic	keyboard	optical touch pannels	optical touch pannels

11	Which is constructed with two transperant plates separated by small distance?	Electrical touch pannels	optical touch pann	Acqustic or sonic	Stegnometer	Electrical touch pannels
12	In __________ high frequency sound waves are generated in horizontal & vertical direction access a glass plates.	Electrical touch pannels	optical touch pann	Acqustic touch pannel	Digital	Acqustic touch pannel
13	_________is an efficient device for inputting such non graphic data as picture labels associated with a graphics display.	Mouse	keyboard	Monitor	CPU	keyboard
14	_________ keys allow users to enter frequently used operations in a single keystroke, and cursor-control keys	Function	Primary	Control	Input	Function
15	_________ is small hand-held box used to position the screen cursor.	keyboard	Monitor	mouse	CPU	mouse
16	__________detects movement across the lines in the grid.	Digital	optical sensor	Graphics	Analog	optical sensor
17	_________ attached to the ball, measure the amount and direction of rotation.	Stegnometer	Graphics monitor	phosphor	Potentiometer	Potentiometer
18	_______are often mounted on keyboards or other devices such as the Z mouse.	Monitor	Trackballs	Mouse	keyboard	Trackballs
19	_______ are used for three-dimensional positioning and selection operations in virtual-reality systems, modeling, animation, CAD,	Trackballs	Mouse	keyboard	Space balls	Space balls
20	A _______consists of a small, vertical lever (called the stick) mounted on a base that is used to steer the screen cursor around.	joystick	Mouse	keyboard	Space balls	joystick
21	_________ is constructed with a series of sensors that detect hand and finger motions.	Mouse	Data glove	keyboard	joystick	Data glove
22	_______coupling between transmitting antennas and receiving antennas is used to provide information about the position and	Potentiometer	Electromagnetic	Phosphor	Magnetic device	Electromagnetic
23	A common device for drawing, painting, or interactively selecting coordinate positions on an object is a _______.	digitizer	keyboard	Data glove	Joystick	digitizer
24	A _____ contains cross hairs for sighting positions, while a stylus is a pencil-shaped device that is pointed at positions on the tablet.	Mouse	keyboard	Glove	hand cursor	hand cursor
25	_____ employ a line of infrared light-emitting diodes	Electrical Touch panels	Acoustic touch panal	Optical touch panels	Magnetic touch panel	Optical touch panels
26	An _______ is constructed with two transparent plates separated by a small distance.	Electrical Touch panels	Acoustic touch panal	Optical touch panels	Magnetic touch panel	electrical touch panel

27	_______ have some advantage over other input devices, since the attention of the operator does not have to be switched from one	Voice systems	Panel	Cursor	Speech system	Voice systems
28	________ press formed character faces against an inked ribbon onto the paper.	Non impact printer	Impact printers	Character impact printer	Inkjet printer	Impact printers
29	_______ and plotters use laser techniques, ink-jet sprays, xerographic processes (as used in photocopying machines),	Non- impact printers	Impact printers	Character impact printer	Inkjet printer	Non- impact printers
30	______ often have a dot-matrix print head containing a rectangular array of protruding wire pins	Non- impact printers	Impact printers	Character impact printer	Inkjet printer	Character impact printers
31	___methods produce output by squirting ink in horizontal rows across a roll of paper wrapped on a drum.	Non- impact printers	Impact printers	Character impact printer	Inkjet printer	Ink-jet
32	__________ device is used to select screen positions by detecting the light coming from points on the CRT screen	keyboard	Data glove	Joystick	Light pens	Light pens
33	To non-zero intensity assigned to each screen ____	keyboard	pixel	Cursor	Speech system	pixel
34	Which input can be used to initiate graphics operations or to enter data	voice system	digitizer	keyboard	Data glove	voice system
35	________ input is typically spoken into a micro phone mounted on a headset	digitizer	voice system	Cursor	Speech system	voice system
36	________ provides 6 degrees of freedom	space ball	voice system	digitizer	keyboard	space ball
37	Space ball is used in ______ system	voice system	Cursor	Speech system	virtual reality	virtual reality
38	________ attach to the ball to measure the amount and direction of rotation	potentio meter	Electromagnetic	Phosphor	Magnetic device	potentio meter
39	_______ is the 2-dimensional positioning device	voice system	track ball	Cursor	Speech system	track ball

KARPAGAM ACADEMY OF HIGHER EDUCATION

(Deemed University Established Under Section 3 of UGC Act 1956)
Coimbatore - 641021.
(For the candidates admitted from 2015 onwards)

DEPARTMENT OF COMPUTER SCIENCE, CA & IT

SUBJECT : COMPUTER GRAPHICS
SEMESTER : V
SUBJECT CODE: 15CSU503 **CLASS : III B.Sc.CS**

UNIT-III

Point and Lines- Line Drawing Algorithms: DDA Algorithm- Bresenhams Line Algorithm. **Circle Generating Algorithms**: Mid Point Circle Algorithm. Two Dimensional Geometric Transformations: **Basic Transformations**: Translation-Rotation-Scaling-**Composite Transformations**: Translations-Rotations- Scaling. General Pivot Point Rotation- General Fixed Point Scaling.

TEXT BOOK

1. Donald Hearn and M. Pauline Baker. 2007. Computer Graphics-C Version, 2nd Edition, Pearson Education, New Delhi.
   (Page Nos. : 24-54, 56-77, 80-92, 103-118, 204-215, 236-256, 427-443, 458-463, 490-495, 502-505)

REFERENCES

1. Amarendra N. Sinha. 2008. *Computer Graphics*, 1st Edition ,Tata McGraw Hill, New Delhi.
2. Foley, Vandam, Feiner and Hughes. 1999. *Computer Graphics Principles and Practices*, 2nd Edition ,Addison Wesley, Singapore.
3. Evangeline D., Anita, S. 2016. *Computer Graphics And Multimedia Insights, Mathematical Models And Programming Paradigms*. PHI Learning Pvt. Ltd.
4. Rae Earnshaw.2014.*Computer Graphics: Developments in Virtual Environments*, 2nd Edition ,academic Press, New Delhi.

WEB SITES:
www.cgshelf.com
www.cgtutorials.com
www.allgraphicdesign.com

UNIT-3

Definition: Lines and points

A line connects two points. It is a basic element in graphics. To draw a line, you need two points between which you can draw a line. In the following three algorithms, we refer the one point of line as $X0,Y0$ and the second point of line as $X1,Y1$.

Line Drawing Algorithms

- Equation of straight line is y= mx + b. where m is the slope of straight line and b is the y intercept.
- M = $\frac{y2-y1}{x2-x1} = \frac{\Delta y}{\Delta x}$
- If $|m| < 1$ then Δx is proportional to Δy.
- If $|m| > 1$ then Δy is proportional to Δx.
- Any change in Δx is called as horizontal deflection, any change in Δy is called as vertical deflection. Δy Or Δx.

Definition: DDA-Digital Differential Analyzer (DDA) algorithm is the simple line generation algorithm which is explained step by step here.

DDA Line Drawing Algorithm

DDA is a scan conversion line algorithm based on calculating either Δy or Δx.We sample the line at unit intervals in one coordinate and determine corresponding integer nearest to the line path for the other coordinates. Line drawing is accomplished by calculating intermediate positions along the line path between two specified end points. Digital devices display a straight-line segment by plotting discrete points between two end points. Discrete coordinates along the line path are calculated from line equations. Screen locations are referenced with integer values; so plotted positions may only approximate actual line positions between two specified endpoints. The rounding of co-ordinate values to integers causes lines to be displayed with a stair step appearance.

To load an intensity value into the frame buffer at a position to column x and line y, the procedure used is

Setpixel (x, y, intensity)

To retrieve the current frame buffer intensity by calling the procedure

Getpixel(x, y)

DDA algorithm is a faster method for calculating pixel positions. The accumulation of round off error in successive additions of the floating-point increment causes the calculated pixel positions to drift away from the true line path for long line segment. The rounding operations & floating-point arithmetic in DDA procedure is time-consuming. The Cartesian slope intercept equation for a straight line is

$$y = m.x+b \rightarrow (1)$$

Here m represents slope of the line and b represents y intercept. Let us consider two end points are (x1, y1) & (x2, y2).

Calculating slope and intercept values

$$m = y2-y1 / x2-x1 \rightarrow (2)$$

$$b = y1-m.x1 \rightarrow (3)$$

$$\Delta y = m.\Delta x \rightarrow (4)$$

$$\Delta x = \Delta y/m \rightarrow (5)$$

Slope magnitudes can set deflection of the voltage.

Lines with the slope magnitude $|m|<1$, Δx have proportional to a small horizontal deflection voltage & vertical deflection proportional to Δy.

$|m|>1$ Δy set to small vertical deflection & horizontal deflection is set to Δx.

$m=1$ $\Delta x=\Delta y$ horizontal and vertical deflection are equal.

DDA algorithm draws lines at unit intervals in one co-ordinate and determines corresponding integer values nearest the line path for the other co-ordinate.

If it is a positive slope

(1) The slope is less than or equal to 1 compute each successive y as

$$Y_{k+1} = Y_k + m$$

Subscript k takes integer value starting from 1 and increase by 1 until the final end point is reached.

(2) the slope is greater than 1 then calculate each succeeding x value as

$X_{k+1} = X_k + (1/m)$

If the process is from right point to left (ie reverse)

$Y_{k+1} = Y_k - m$

$X_{k+1} = X_k - (1/m)$

DDA algorithm accepts as a input the two end point pixel positions. Find Horizontal and vertical differences between endpoints positions are assigned to parameters dx and dy. The difference with greater magnitude determines the value of parameter steps. Starting with pixel position (xa, ya), generate next pixel position along the line path. Loop through this process steps times. The value of x and y will get incremented by using algorithm calculations. Then call setpixel for plot pixel. DDA algorithm is faster method. The following algorithm explains the line drawings.

```
#include “device.h”
#define ROUND (a) ((int) (a+0.5))
void lineDDA (int xa, int ya, int xb, int yb)
{
  Int dx=xb-xa, dy=yb-ya, steps, k;
  float xIncrement, yIncrement, x=xa, y=ya;
        if ( abs(dx) > abs(dy) ) then steps :=  abs(dx);
          else steps := abs(dy);
          xIncrement := dx / (float) steps;
          yIncrement := dy / (float) steps;
          setPixel  (ROUND(x) , ROUND(y) );
          for (k := 0;k<steps; k++) {
                  x := x + xIncrement;
                  y := y + yIncrement;
                setPixel  (ROUND(x) , ROUND(y) );
          }
 }
```

Disadvantage: - round of function – drift away from the true line path for long line segment.

Bresenham’s Line Drawing Algorithm:

An accurate and efficient raster line generating algorithm, developed by Bresenham, scan converts lines only incremental integer calculation that can be adapted to display circles and other curves. The vertical axes show scan-line positions, and the horizontal axes identity pixel columns. Sampling at unit x intervals in these intervals, we need to find the next closest pixel position in the line, whose value is proportional to difference between the separations of the two pixel positions from the actual path.

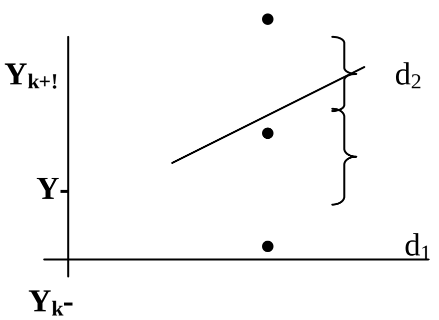
Starting from the left endpoint (10,11) we need to determine at the next sample position whether to plot at position (11, 11) or the one at (11, 12).These questions are answered with Bresenham's line algorithm by testing the sign of an integer parameter, whose value is proportional to the difference between the separations of the two pixel positions from the actual line path.

Consider the scan conversion process for lines with positive slope less than 1.Pixel positions along line path are then determined by sampling at unit x intervals. Starting from left end point(xo,yo) of a given line, we step to each successive column and plot pixel whose scan line value is closest to the line path. Assume we have determined that the pixel ($x_{k,}y_k$) is to be displayed. Next has to decide which pixel to plot in column x_{k+1}.

The line equation is $y= m(x_k +1)+b$ →eq1)

d1 = $y-y_k$ =→ $m(x_k +1)+b-y_k$ // implemented in eq1

d2 = $(y_k+1)-y$ =→ $y_k+1-m(x_k +1)-b$// implemented in eq1.



The differences between these two separators are

$$M = \Delta y/\Delta x$$

$$d1 - d2 = 2m(x_k + 1) - 2y_k + 2b - 1$$

A decision parameter P_k for k th step.

$$P_k = \Delta x(d1 - d2)$$

The first parameter can be calculated by the equ $p_0 = 2\Delta y - \Delta x$.

Next point can calculated by $p_{k+1} = p_k + 2\Delta y$ and $p_{k+1} = p_k + 2\Delta y$.

Bresenham's Line – Drawing Algorithm

- Input the two line endpoints and store the left endpoint in (x_0 , y_0).
- Load (x_0, y_0) into the frame buffer; that is plot the first point.
- Calculate constants Δx, Δy, $2\Delta y$, and $2\Delta y - 2\Delta x$, and obtain the starting value for the decision parameter as

  $$p_0 = 2\Delta y - \Delta x$$
- At each x_k along the line, starting at k = 0, perform the following test:

  If $p_k < 0$, the next point to plot is $(x_k + 1, y_k)$ and

  $$p_k + 1 = p_k + 2\Delta y$$

  Otherwise, the next point to plot is $(x_k + 1, y_k + 1)$ and

  $$p_k + 1 = p_k + 2\Delta y - 2\Delta x$$
- Repeat step 4 Δx times.

```
#include "device.h"
void lineBres (int xa, int ya, int xb, int yb)
{
 int dx = abs(xa – xb), dy = abs(ya – yb);
 int p = 2 * dy – dx;
 int twody = 2* dy, twody = 2 * (dy-dx);
 int x, y, xEnd;
   /* determine which point to use as start, which as end */
               if xa  > xb then{
                   x = xb;
```

```
y = yb;
   xEnd = xa;}
  else{
        x = xa;
        y = ya;
   xEnd = xb;}
 setPixel  ( x, y);
 while (x <  xEnd) {
        x = x ++;
        if p < 0 then p += twoDy;
        else{
              y ++;
              p += twoDyDx;
        }
           setPixel ( x,  y)
 }
}
```

<u>Circle Generation Algorithm</u>

<u>Properties of a circle</u>:

A circle is defined as the set of points that are all at a given distance r from a center position (x_c , y_c).

Pythogorean theorm in caretisan coordinates as $(x-x_c)^2+ (y-y_c)^2= r^2$

By using this calculate position of points on a circle circumference by stepping along the x axis in unit steps from x_c -r to x_c +r and calculate y as

Y= y_c + or –sqrt(r_2-(x_c –r$)_2$)

Problem in this approach

1. Involves more calculations at each step.
2. Spacing between plotted pixel position is not uniform.

To eliminate unequal spaces by polar coordinate as

$x = x_c + r\cos\theta$

$y = y_c + r\sin\theta$

when a display is generated with these equations using a fixed angular step size, a circle is plotted with equally spaced points along the circumference. θ depends on the application and display device.

Shape of the circle is similar in each quadrant. One quadrant are symmetric with respect to the 45°

More efficient circle algorithm are based on incremental calculation of decision parameter. Bresenhams algorithm method for direct distance comparison to test the halfway position between the two pixel to determine this midpoint is inside or outside the circle boundary.

Circle function for calculating mid point is $f_{circle}(x,y) = x^2+y^2-r^2$

$F_{circle}(x,y)$

<0 if (x,y) is inside the circle boundary.

=0 if(x,y) is on the circle boundary

>0 if(x,y) is outside the boundary.

Let we consider circle start position (x0,y0)=(0,r).

$P_0 = f_{circle}(1,r-1/2)$

$P_0 = 5/4-r$

$P_0 = 1-r.$

Midpoint Circle Algorithm

- Input radius r and circle center (x_c,y_c), and obtain the first point on the circumference of a circle centered on the origin as

  $$(x_0,y_0) = (0,r)$$

- Calculate the initial value of the decision parameter as

  $$p_0 = 5/4 – r$$

- At each x_k, position, starting at k = 0, perform the following test : If $p_k < 0$, the next point along the circle centered on (0,0) is (x_{k+1}, y_k) and

  $$p_{k+1} = p_k + 2x_{k+1} + 1$$

  Otherwise, the next point along the circle is (xk + 1, yk –1) and

  $$p_{k+1} = p_k + 2x_{k+1} + 1 – 2y_{k+1}$$

  Where $2x_{k+1} = 2x_k + 2$ and $2y_{k+1} = 2y_k – 2$.

- Determine symmetry points in the other seven octants.
- Move each calculated pixel position (x,y) onto the circular path centered on (x_c, y_c) and plot the coordinate values:

$x = x + x_c$, $y = y + y_c$

- Repeat steps 3 through 5 until x >= y;

```
Include"device.h"
void circleMidpoint (int xCenter, int yCenter, int radius);
{
  int  x=0, y=radius;
    int p=1-radius;
void circleplotPoints(int, int, int , int )
/* float first set of points */
Circlefloatpoints(Xcenter, Ycenter, x,y)
While(x<y)
{
    x++;
if (p<0)
      p+=2*x+1;
else
 {
      y--;
      p := p + 2 * (x – y) + 1;
 } }
void circle plotpoints(int xCenter, int yCenter, int x, int y)
{
          setPixel (xCenter + x, yCenter + y);
          setPixel (xCenter - x, yCenter + y);
          setPixel (xCenter + x, yCenter - y);
          setPixel (xCenter - x, yCenter - y);
          setPixel (xCenter + y, yCenter + x);
          setPixel (xCenter - y, yCenter + x);
          setPixel (xCenter + y, yCenter - x);
          setPixel (xCenter - y, yCenter - x);
      }
```

TWO DIMENSIONAL GRAPHICS:

BASIC TRANSFORMATION:

1. TRANSLATION.
2. ROTATION.
3. SCALING.

1. <u>TRANSLATION</u>: -

- It is applied to an object by repositioning it along a straight line from one coordinate location to another.
- We translate a 2-D point by adding translation distances, tx & ty to the original coordinate position (x, y) to move the point to a new position (x', y').
- **x' = x +tx & y' = y + ty.**
- The translation distance pair (tx , ty) is called a Translation vector or Shift vector.
- Translation equations as a single matrix equations by column vectors represent the coordinates:

  P' = P +T. → (1)
- Where **P=** $\begin{bmatrix} x1 \\ x2 \end{bmatrix}$, **P' =** $\begin{bmatrix} x1' \\ x2' \end{bmatrix}$, **T =** $\begin{bmatrix} tx \\ ty \end{bmatrix}$
- In terms of coordinate row vectors :

  P = $[x, y]$ & T = $[tx, ty]$
- Translation is a Rigid-body transformation that moves object without deformation.
- A straight-line segment is translated by applying the transformation equation (1) to each of the line endpoints and redraws the line between the new endpoint positions.
- Polygons are translated by adding the translating vector to the coordinate position of each vertex and regenerating the polygons using the new set of vertex coordinates.
- To change the position of a circle or ellipse we translate the center coordinates and redraw the figure in the new location.
- If tx, ty value is higher than the width value then there will be an error [Wraparound].

2. <u>ROTATION:</u>

- A 2-D rotation is applied to an object by repositioning it along a circular path in the XY plane.
- To generate a rotation specify a rotation point (or) pivot point about which the object is to be rotated.

- Positive values for rotation angle define counterclockwise rotation about the pivot point.
- Negative values rotate object in the clockwise direction.
- This transformation can also be described as a rotation about a rotation axis that is perpendicular to the XY plane and passes through the pivot point.

1. The transformation equation for rotation of a point position P when the pivot point is at the coordinate origin:

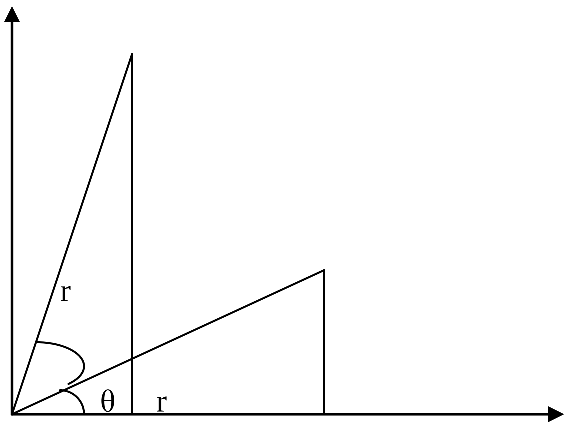


Figure shows the angular and coordinate relationships of the original and transformed point positions:

- In the figure r is the constant distance of the point from the origin.
- Angle Φ is the original angular position of the point from the horizontal.
- Θ is the rotation angle.
- By using the trigonometric identities;

$$\cos \Theta = \frac{adj}{hyp}$$

$\sin \Theta = \frac{opp}{hyp}$

$\tan \Theta = \frac{opp}{adj}$

$X = r \cos \Phi \rightarrow (1)$

$Y = r \sin \Phi \rightarrow (2)$

$X' = r \cos (\Phi + \Theta) \rightarrow (3)$

$Y' = r \sin (\Phi + \Theta) \rightarrow 4)$

2. <u>Rotation of a point about an arbitrary pivot position:</u>

   Objects can be rotated about an arbitrary point by modifying the equation (7) to include the coordinates (xr , yr) for the selected rotation point.

   - The transformation equations for the rotated coordinates are obtained by the trigonometric relationship.

     $X' = xr + (x - xr) \cos\Theta - (y-yr) \sin\Theta$

     $Y' = yr + (x - xr) \sin\Theta + (y - yr) \cos\Theta$

   - Every point on an object is rotated through the same angle.

3. SCALING:

   - Scaling transformation alters the size of an object.
   - For polygon the scaling can be carried out by multiplying the coordinate values (x , y) of each vertex by scaling factors sx & sy to produce the transformed coordinates (x', y') :

     $x' = x.sx$ & $y' = y.sy \rightarrow (1)$

   - Scaling factors sx scales object in the x-direction & sy in the y-direction.

   - Transformation equation in the matrix form :

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} sx & 0 \\ 0 & sy \end{bmatrix} . \begin{bmatrix} x \\ y \end{bmatrix}$$

p' =s.p

- Any positive values can be assigned to the scaling factors sx & sy.
- Sx & sy values less than 1 reduce the size of objects.
- Values greater than 1 produces enlarged object.
- Value of sx & sy is 1means leaves the size of objects unchanged.
- When sx & sy are of same value a uniform scaling is produced.
- Unequal values for sx & sy results in a differential scaling.
- We can control the location of a scaled object by choosing a position called the fixed point, which remains unchanged after the scaling transformation.
- Scaling relative to a chosen fixed point (xf , yf) is:

$$\begin{Bmatrix} X = xf + (x - xf).\, Sx \\ y = yf + (y - yf).\, sy \end{Bmatrix} \rightarrow \textbf{(3)}$$

(3) Same as

$$\begin{Bmatrix} x' = x.sx + (1 - sx).xf \\ y' = y.sy + (1 - sy).yf \end{Bmatrix}$$

MATRIX REPRESENTATIONS & HOMOGENEOUS COORDINATES:

- To express any 2-D transformation as a matrix multiplication we represent each Cartesian coordinate position (x, y) with the homogeneous coordinate triple (xh, yh, h) where X= $\frac{xh}{h}$ Y= $\frac{yh}{h}$
- Homogeneous coordinate can also be represent as (xh, yh, h).
- For 2-D geometric transformation we choose the homogeneous parameter h to be any non-zero value.
- For 2-D homogeneous coordinates (x, y, 1).
- Other values for parameter h are needed for 3-D viewing transformation.

For translation transformation:

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & tx \\ 0 & 1 & ty \\ 0 & 0 & 1 \end{bmatrix} . \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

P'=T (x, y). P

X'=x + tx

Y'=y + ty

1=1

For Rotation Transformation:

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} . \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

P'=R (θ). P

X'=x. Cosθ -y. Sinθ

Y'=y Cosθ +x. Sinθ

For Scaling Transformation:

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} sx & o & o \\ 0 & sy & 0 \\ 0 & 0 & 1 \end{bmatrix} . \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

P'=S (Sx, Sy). P

X'=x. Sx

Y'=y. Sy

- Matrix representations are standard methods for implementing transformation in graphics system.

COMPOSITE TRANSFORMATION:

- It means calculating the matrix product of the individual transformations.
- The resultant matrix is referred to as a concatenation or composition of matrices.

1. **Translations:**

$$\begin{bmatrix} 1 & 0 & tx2 \\ 0 & 1 & ty2 \\ 0 & 0 & 1 \end{bmatrix} . \begin{bmatrix} 1 & 0 & tx1 \\ 0 & 1 & ty1 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & tx1+tx2 \\ 0 & 1 & ty1+ty2 \\ 0 & 0 & 1 \end{bmatrix}$$

T (tx2, ty2). T (tx1, ty1) = T (tx1+tx2, ty1+ty2)

2. **Rotations:**

   P' =R ($\theta 1 + \theta 2$) .P

3. **Scaling:**

$$\begin{bmatrix} sx2 & 0 & 0 \\ 0 & sy2 & 0 \\ 0 & 0 & 1 \end{bmatrix} . \begin{bmatrix} sx1 & 0 & 0 \\ 0 & sy1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} sx1.sx2 & 0 & 0 \\ 0 & sy1.sy2 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

S (sx2, xy2). S (sx1, sy1)=S (sx1.sx2, sy1.sy2)

4. **General Pivot-Point Rotation:**
   - Rotation about any selected pivot-point (x, y) by performing translation-rotation-translation.
   - **Steps:**
     a) Translate the object so that the pivot point position is moved to the coordinate origin.
     b) Rotate the object about the coordinate origin.
     c) Translate the object so that the pivot point is returned to its original position.

5. **General Fixed Point Scaling:**
   a) Translate objects so that the fixed point coincides with the coordinate origin.
   b) Scale the object with respect to the coordinate origin.
   c) Use the inverse translation of step1 to return to the original position.

OTHER TRANSFORMATION:

i.Reflection:

- A reflection is a transformation that produces a mirror image of an object.

- The mirror image for 2-D reflection is generated relative to an axis of reflection by rotating the object 180* about the reflection axis.

   i. Reflection about the line y=0 the x-axis is accomplished with the transformation matrix:

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & \\ 0 & 0 & 1 \end{bmatrix}$$

This transformation keeps x-values the same but flips the y values of coordinate position.

ii. Reflection about the line x=0 the y-axis is accomplished with the transformation matrix.

$$\begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

In this x-coordinates are flipped and y coordinates are same.

iii. If we flip both x and y coordinates of a point by reflecting relative to an axis that is perpendicular to the xy plane and that through the coordinate origin. Matrix representation:

$$\begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

iv. Reflection axis as the diagonal line y=x the reflection matrix is

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Steps:

1) First perform a clockwise rotation through a 45* angle which rotates the line y=x onto the x-axis.
2) Next perform a reflection with respect to the x-axis.

3) Finally rotate the line y=x back to its original position with a counter clock wise rotation through 45*.

Another equivalent steps :

1) First reflect the object about the x-axis.
2) Then to rotate counter clockwise 90*.

[Figure & Derivation Refer Class Notes

Transformation matrix for reflection about the diagonal y= -x.

$$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

<u>Steps:</u>

1) Clockwise rotation by 45*
2) Reflection about y-axis.
3) Counter clockwise rotation by 45*

<u>Another Format</u>

1) Reflect about y-axis.
2) Rotate Counter Clockwise 90*.

<u>Shear:</u>

- A transformation that distorts the shape of an object such that the transformed shape appears as if the object were composed of internal layers that had been caused to slide over each other is called a SHEAER.
- A x-direction shear relative to the x-axis:

$$\begin{bmatrix} 1 & shx & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

X value is changed & y value remains same.

- A y-direction shear relative to the line y-axis:

$$\begin{bmatrix} 1 & 0 & 0 \\ shy & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Y value is changed & x value remains constant.

POSSIBLE QUESTIONS
UNIT – III

Part-A

Online Examinations (One marks)

1. Picture definition is stored in a memory area is called _______
   a) Memory b) Picture definition c) **Frame Buffer** d) Fluorecent
2. The ________ in CRT is needed to force the electron beam to converge into a small spot as it strikes the phosphor.
   a) **Focusing system** b) Control grid c) Accessing point d) Pixel Position
3. Deflection of electron beam can be controlled either with______
   a)Potentiometer b) Magnetic diode c)Pixel or Pel
   d)**Electric field or magnetic field.**
4. The maximum number of points that can be displayed without overlapping on a CRT is referred as the ______
   a) Picture definition b) **Resolution** c) Magnetic field d) None of the Above
5. Another method for detecting mouse motion is
   a)Primary gunb) Control grid c)Input d)**optical sensor**
6. The tablet use sound waves to detect a system position
   a) Graphics position b) **Acoustic or sonic** c)Stegnometer d)Graphics monitor
7. Three dimensional digitizers use sonic or _______transmission to record positions
   a) Electrical Touch panels b) Acoustic touch panel c) Phosphor
   d) **Electromagnetic**
8. Which dimensional digitizer designed for transmission to apple Macintosh computers
   a) **Three dimensional** b) keyboard c) Space balls d) two dimensional
9. _________ attach to the ball to measure the amount and direction of rotation
   a)**potentiometer** b) Electromagnetic c)Phosphor d)Magnetic device
10. _______ is the 2-dimensional positioning device
    a) Voice system b) **track ball** c) Cursor d) Speech system

Part B

Essay Type (8 Marks)

1. Write down the algorithm for Brenham's Line Drawing Algorithm.
2. What is shearing. Explain in detail?
3. What is the need of homogeneous coordinates?
4. Explain midpoint subdivision algorithm?
5. What is persistence?
6. What is rotation?
7. What are the disadvantages of DDA algorithm?
8. What is translation?
9. List out the Basic Transformation.

12. Enumerate the Bresenham's Circle Generating algorithm.
13. Explain in detail about Basic Transformation.
14. Write a procedure for DDA line drawing algorithm.
15. Explain the transformations-reflection and shearing with examples.
16. Explain Bresenhams Line Drawing Algorithm with Example.
17. Derive decision parameter for the midpoint circle algorithm.
18. Explain briefly about the other transformations?

KARPAGAM ACADEMY OF HIGHER EDUCATION

COIMBATORE - 21

DEPARTMENT OF COMPUTER SCIENCE

CLASS : III B.Sc COMPUTER SCIENCE

BATCH : 2015-2018

Part -A Online Examinations **(1 mark questions)**

SUBJECT: COMPUTER GRAPHICS **SUBJECT CODE: 15CSU503**

	UNIT III					
S.No	**Questions**	**option1**	**option2**	**option3**	**option4**	**Answer**
1	_____ is accomplished by calculating intermediate positions along the line path between two specified endpoint positions.	DDA	Parellel Lines	Line drawing	Bresenham's	Line drawing
2	________ display a straight line segment by plotting discrete points between the two endpoints.	Digital devices	Parellel Lines	Analog device	Points	Digital devices
3	_________ is a scan-conversion line algorithm.	Line drawing	DDA	Parellel lines	Bresenham's	DDA
4	A random scan system stores point-plotting instruction in the	Frame buffer	display list	Refresh buffer	picture definition	display list
5	Digital devices display a straight line segment by plotting discrete points between the two ________	end points	Electron beam	Electron guns	Control grid	end points
6	Screen locations are referred with ____________	Parellel lines	Digital Differen	integer value	parameters	integer value
7	To load an intensity value into the frame buffer at a position corresponding to column x along scan line y,we have available a ___of the form	low level procedure	High level procedures	Cartesian Slope	Deflection slope	low level procedure

8	The cartesian slope-intercept equation for a straight line is	Y or y + 1	y=m,y+5	x+5	y=m.x+6	y=m.x+6
9	On raster systems lines are plotted with ________	pixels	points	lines	parellel	pixels
10	For lines with slope magnitude|m|<1,Δx can be set proportional to a small	vertical delflection	positive slope	negative slope	horizontal deflection voltage	horizontal deflection voltage
11	For lines with slope magnitude|m|>1,Δy can be set proportional to a small	low level voltage	vertical deflection voltage	high level voltage	horizontal deflection voltage	vertical deflection voltage
12	For lines with m=1,Δx=Δy and the horizontal and vertical deflection voltage are	equal	not equal	infinity	null	equal
13	To retrieve the current frame-buffer intensity setting for a	High level procedures	Cartesian Slope	Deflection slope	specified location	specified location
14	DDA stands for ____________	Data Differential Analyzer	Data devise analysis	Digital Differential algorithms	Digital Differential Analyzer	Digital Differential Analyzer
15	Horizontal and vertical difference between the endpoint positions are assigned to	Arguments	Null values	parameters	P	parameters
16	If the magnitude of dx is greater than the magnitude of dy and xa is less than xb,the values of the increments in the x and y directions are	X and y	l and m	x and z	r and m	l and m
17	If the greater change is in the x direction but xa is greater than xb,then the directions ____________ are used to generate each new point on the line	#NAME?	#name	Name?	name?	#NAME?
18	The DDA algorithm is faster method for calculating_______ position	Points	Lines	Rows points	pixel	pixel

19	The performance of the DDA algorithm by separating the increaments m and 1/m into parts	integer and fractional	scan line positions	functional line	slope of the curve = -1	integer and fractional
20	A decision parameter Pk for the kth step in the line algorithm can be obtained by	normalized	tranformal	direct	rearranging	rearranging
21	The vertical axes show ________________	pixel position	horizontal position	scan line positions	raster line	scan line positions
22	The horizontal axes identify ___________	scan line	pixel columns	vertical lines	None of the above	pixel columns
23	Pixel positions along a line path are then determine by sampling at unit	x intervals	y intervals	x and y	x-y	x intervals
24	An implementation of bresenham line drawing for slope in the range _____________ given in the following procedure	m>1	0<m<1	m<1	0>m>1	0<m<1
25	The call to _______ loads the intensity value 1 into the frame buffer at the specified (x,y)	get pixel	lines	set pixel	buffering	set pixel
26	________ is the frequently used component in pictures and graphs	lines	circle	semi circle	parellel	circle
27	If the point is outside the circle,the circle funtion is	positive	Pre-filtering	Area filtering	Super filtering	positive
28	The circle funtion is the decision parameter in the	radius	center	midpoint algorithm	drawing algorithm	midpoint algorithm
29	Some monitors use a technique called ____________ to double their refreshing rate.	flickering	interlacing	doubling	persistence	interlacing

30	Bresenham’s circle generating algorithm will take reflections of ________	Two octets	One octet	Three octets	Four octets	One octet
31	Eight-way symmetry is used by reflecting each calculated point around each ___ axis	35°	180°	45°	90°	45°
32	In Bresenham’s circle generating algorithm, if(x,y) is the current pixel position then the x-value of the next pixel position is	X	X – 1	X + 1	X + 2	X + 1
33	In Bresenham’s circle generating algorithm, if(x,y) is the current pixel position then the y-value of the next pixel position is	Y or y + 1	Y alone	Y + 1 or y - 1	Y or y – 1	Y or y – 1
34	The property that adjacent (neighbouring) pixels are likely to have the same characteristics is called	Area coherence	Spatial coherence	Scan line coherence	Pixel coherence	Spatial coherence
35	The property that adjacent pixels on a scan line are likely to have the same characteristics is called	Area coherence	Spatial coherence	Scan line coherence	Pixel coherence	Scan line coherence
36	Filling polygons can be done by setting the pixels	Inside, outside	Inside, boundary	Outside, boundary	In-between	Inside, boundary
37	The design style of set of character is referred to as its	Typeface	Font size	Font style	None of the above	Typeface
38	Character sizes approximately ranges to	1/12 inch	¾ inch	2/5 inch	½ inch	1/12 inch
39	The technique of using a minimum number of intensity levels to obtain increased visual resolution is called	Dithering	Depth cueing	Rendering	Halftoning	Halftoning
40	In Bresenham’s line generating algorithm, the decision variable is _	D(T)+D(S)	Dx(s-t)	Both a and b	None of the above	Dx(s-t)

41	In Bresenham's circle generating algorithm, the decision variable is _	D(T)+D(S)	Dx(s-t)	Both a and b	None of the above	D(T)+D(S)
42	Character sizes approximately ranges to	10 point	11 points	12 points	14 points	12 points
43	A _______ is a small raster containing the relative locations of the pixels that are used to represent the character	Frame buffer	Mask	Display	Shadow	Mask
44	In Line display of characters ________ test is used to determine the intersecting lines	Integration	Black box	Unit	Minmax	Minmax
45	When ____________ is used for a picket fence problem, the distance between pickets are kept close to their true distance	Private aliasing	local aliasing	Global aliasing	Public aliasing	local aliasing
46	When ____________ is used for a picket fence problem, the overall length of the picket fence is approximately correct	Private aliasing	local aliasing	Global aliasing	Public aliasing	Global aliasing
47	__________ occurs when an object is not aligned with, or does not fit into, the pixel grid properly.	Staircase	Picket Fence problem	Unequal Brightness	Asymmetric location	Picket Fence problem
48	A ___________ technique works on the true signal in the continuous space to derive proper values for individual pixels	Post-filtering	Pre-filtering	Area filtering	Super filtering	Pre-filtering
49	A ___________ technique takes discrete samples of the continuous signal and uses the samples to computer pixel values	Post-filtering	Pre-filtering	Area filtering	Super filtering	Post-filtering
50	In using methods like Bresenhams algorithm to scan convert arcs there is a danger of missing the end points of the arc. This may result in	infinite loop	ellipse	staircase effect	floating point calculation	infinite loop
51	_____ is a arc with two lines drawn from the center to the end points of the arc.	Circle	Sector	Ellipse	Curve	Sector

52	Ellipse exhibits _________ way symmetry.	two	four	six	eight	four
53	Polynomial method of scan converting an ellipse is inefficient because it uses _________	floating addition and subtraction	logarithmic calculations	square and square root operations	integration and differentiation	square and square root operations
54	In Midpoint algorithm, we split the ellipse into two parts at a point Q where _________	major axis = minor axis	x and y coordinate values are	x and y coordinate values are	slope of the curve = -1	slope of the curve = -1
55	In midpoint ellipse algorithm, if the decision parameter is greater than zero the midpoint is the curve	inside	outside	equidistant from	None of the above	outside
56	In midpoint ellipse algorithm, if the decision parameter p< 0 the midpoint is _______ the curve	inside	outside	equidistant from	None of the above	inside
57	For scan converting of ellipse, the first part of the curve the x and y values are obtained by	x is incremented and y is decremented	y is decremented and x is chosen	x is incremented and y is chosen the point close	both x and y are incremented	x is incremented and y is chosen the point close to
58	For scan converting of ellipse, the second part of the curve the x and y values are obtained by	x is incremented and y is decremented	y is decremented and x is chosen	x is incremented and y is chosen the point close	both x and y are incremented	y is decremented and x is chosen the point close to
59	In trigonometrically defining the ellipse x= a cos θ + h and y = b sin θ + k; then the point (h,k) is the of the ellipse.	major axis	minor axis	center	point on the curve	center

KARPAGAM ACADEMY OF HIGHER EDUCATION

(Deemed University Established Under Section 3 of UGC Act 1956)
Coimbatore - 641021.
(For the candidates admitted from 2015 onwards)
DEPARTMENT OF COMPUTER SCIENCE, CA & IT

SUBJECT : **COMPUTER GRAPHICS**
SEMESTER : **V**
SUBJECT CODE : **15CSU503** **CLASS** : **III B.Sc.CS**

UNIT-IV

Two Dimensional Viewing: The Viewing Pipeline- Window to view port Transformation-**Clipping Operations**-Point Clipping -Line Clipping: Cohen Sutherland Line Clipping. Polygon Clipping: Sutherland –Hodgeman Polygon Clipping-Text Clipping.

TEXT BOOK

1. Donald Hearn and M. Pauline Baker. 2007. *Computer Graphics-C Version*, 2nd Edition, Pearson Education, New Delhi.
   (Page Nos. : 24-54, 56-77, 80-92, 103-118, 204-215, 236-256, 427-443, 458-463, 490-495, 502-505)

REFERENCES

1. Amarendra N. Sinha. 2008. *Computer Graphics*, 1st Edition ,Tata McGraw Hill, New Delhi.
2. Foley, Vandam, Feiner and Hughes. 1999. *Computer Graphics Principles and Practices*, 2nd Edition ,Addison Wesley, Singapore.
3. Evangeline D., Anita, S. 2016. *Computer Graphics And Multimedia Insights, Mathematical Models And Programming Paradigms*. PHI Learning Pvt. Ltd.
4. Rae Earnshaw.2014.*Computer Graphics: Developments in Virtual Environments*, 2nd Edition ,academic Press, New Delhi.

WEB SITES:

www.cgshelf.com
www.cgtutorials.com
www.allgraphicdesign.com

UNIT-IV

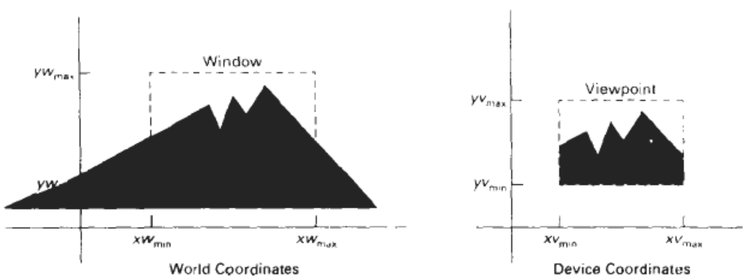
Two dimensional viewing

We now consider the formal mechanism for displaying views of a picture on an output device. Typically, a graphics package allows a user to specify which part of a defined picture is to be display& and where that part is to be placed on the display device. Any convenient Cartesian coordinate system, referred to as the world-coordinate reference frame, can be used to define the picture. For a two-dimensional picture, a view is selected by specifying a sub area of the total picture area. A user can select a single area for display, or several areas could be selected for simultaneous display or for an animated panning sequence across a scene. The picture parts within the selected areas are then mapped onto specified areas of the device coordinates. When multiple view areas are selected, these areas can be placed in separate display locations, or some areas could be inserted into other, larger display areas. Transformations from world to device coordinates involve translation, rotation, and scaling operations, as well as procedures for deleting those parts of the picture that are outside the limits of a selected display area.

The Viewing pipeline

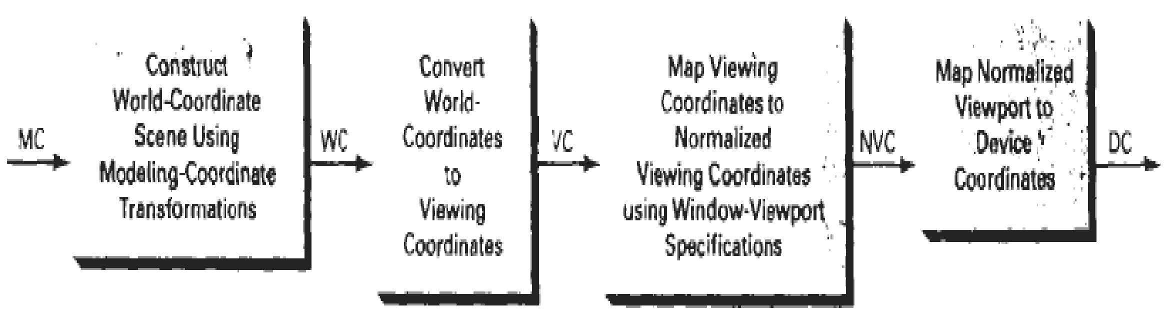
A world-coordinate area selected for display is called a window. An area on a display device to which a window is mapped is called a view port. The window defines what is to be viewed; the view port defines where it is to be displayed. Often, windows and view ports are rectangles in standard position, with the rectangle edges parallel to the coordinate axes. Other window or view port geometries, such as general polygon shapes and circles, are used in some applications, but these shapes take longer to process. In general, the mapping of a part of a world-coordinate scene to device coordinates is referred to as a viewing transformation. Sometimes the two-dimensional viewing transformation is simply referred to as the window-to-view port transformation or the windowing transformation. But, in general, viewing involves more than just the transformation from the window to the view port. Figure1 illustrates the mapping of a picture section that falls within a rectangular window onto a designated &angular view port.

In computer graphics terminology, the term window originally referred to an area of a picture that is selected for viewing, as defined at the beginning of this section. Unfortunately, the same tern is now used in window-manager systems to refer to any rectangular screen area that can be moved about, resized, and made active or inactive. In this chapter, we will only use the term window to refer to an area of a world-coordinate scene that has been selected for display.



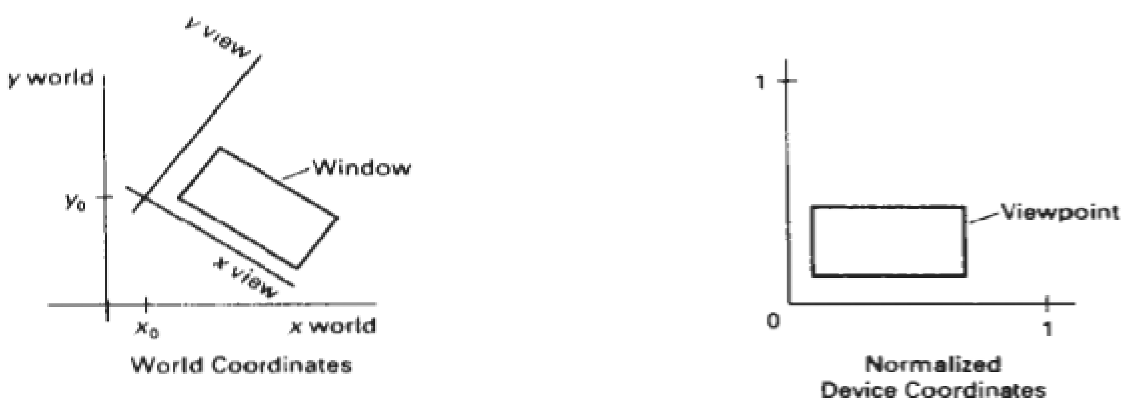
A viewing transformation using standard rectangle for the windows and view port.

Some graphics packages that provide window and view port operations allow only standard rectangles, but a more general approach is to allow the rectangular window to haw any orientation.



The 2D viewing transformation pipeline

In this case, we carry out the viewing transformation in several steps, as indicated in Figure First, we construct the scene in world coordinates using the output primitives and attributes. Next. to obtain a particular orientation for the window, we can set up a two-dimensional viewing-coordinate system in the world-coordinate plane, and define a window In the viewing-coordinate system. The viewing coordinate reference frame is used to provide a method for setting up arbitrary orientations for rectangular windows. Once the viewing reference frame is established, we can transform descriptions in world coordinates to viewing coordinates. We then define a view port in normalized coordinates (in the range from 0 to 1) and map the viewing-coordinate description of the scene to normalized coordinates. At the final step, .all parts of the picture that he outside the view port are clipped, and the contents of the view port are transferred to device coordinates. Figure, i1lustratt.s a rotated viewing-coordinate reference frame and the mapping to normalized coordinates.



Setting up rotated world windows in viewing coordinates and the corresponding normalized – coordinate view port.

By changing the position of the view port, we can view objects at different positions on the display area of an output device. Also, by varying the size of view ports, we can change the size and proportions of displayed objects. We achieve zooming effects by successively mapping different-sized windows on a fixed-size view port. As the windows are made smaller, we zoom in on some part of a scene to view details that are not shown with larger windows. Similarly, more overview is obtained by zooming out from a section of a scene with successively larger windows. Panning effects are produced by moving a fixed-size window across the various objects in a scene.

View ports are typically defined within the unit square (normalized coordinates). This provides a means for separating the viewing and other transformations from specific output-device requirements, so that the graphics package is largely device-independent. Once the scene has been transferred to normalized coordinates, the unit square is simply mapped to the display area for the particular output device in use at that time. Different output devices can be used by providing the appropriate device drivers.

Windows to view port coordinate transformation.

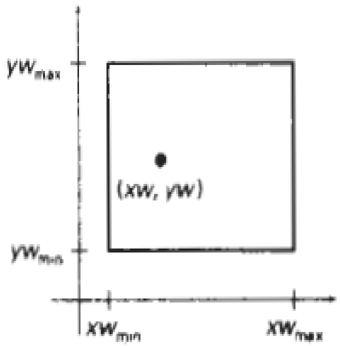
Once object descriptions have been transferred to the viewing reference frame, we choose the window extents in viewing limits in normalized coordinates. Object descriptions are then transferred to normalized device coordinates. We do this using a transformation that maintains the same relative placement of objects in normalized space as they had in viewing coordinates. If a coordinate position is at the center of the viewing window, for instance, it will be displayed at the center of the view port.

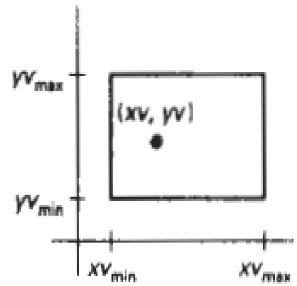
Figure4 illustrates the window-to-view port mapping. A point at position (xw, yw) in the window 1s mapped into position *(xv,* yv) in the associated view port. To maintain the same relative placement in the view port as in the window, we require that

$$\frac{xv - xv_{min}}{xv_{max} - xv_{min}} = \frac{xw - xw_{min}}{xw_{max} - xw_{min}}$$

$$\frac{yv - yv_{min}}{yv_{max} - yv_{min}} = \frac{yw - yw_{min}}{yw_{max} - yw_{min}}$$

Fig4: A point at position (xw, yw) in redesigned windows is mapped to view port coordinates (xv, yv) so that relative position in the two areas are the same.





Solving these expressions for the view port position (xv, yv), we have ,

$$xv = xv_{min} + (xw - xw_{min})sx$$
$$yv = yv_{min} + (yw - yw_{min})sy$$

where the scaling factors are

$$sx = \frac{xv_{max} - xv_{min}}{xw_{max} - xw_{min}}$$

$$sy = \frac{yv_{max} - yv_{min}}{yw_{max} - yw_{min}}$$

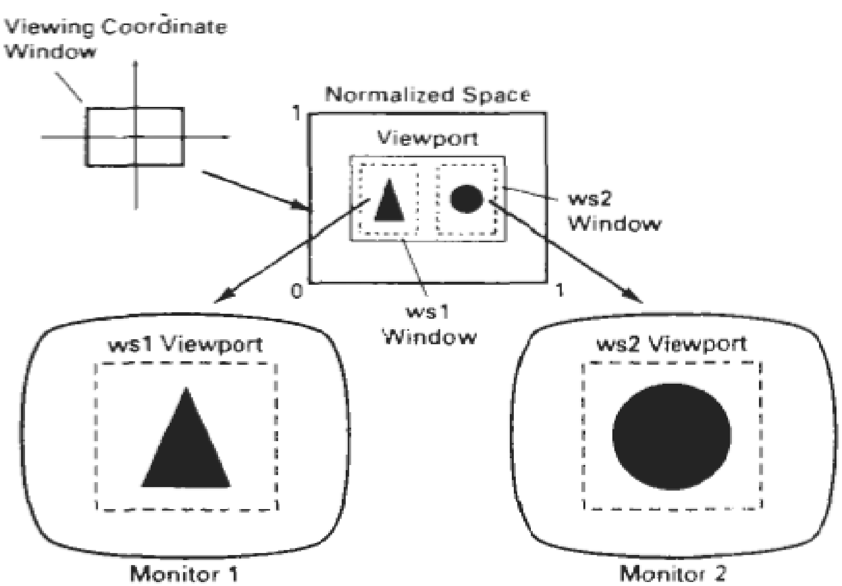
Equations can also be derived with a set of transformations that converts the window area into the view port area. This conversion is performed with the following sequence of transformations:

1. Perform a scaling transformation using a fixed-point position of (xw_{min}, yw_{min}) that scales the window area to the size of the view pot.
2. Translate the scaled window area to the position of the view port.

Relative proportions of objects are maintained if the scaling factors are the same *(sx = sy).* Otherwise, world objects will be stretched or contracted in either the *x* or *y* direction when displayed on the output device.

Character strings can be handled in two ways when they are mapped to a view port. The simplest mapping maintains a constant character size, even though the view port area may be enlarged or reduced relative to the window. This method would be employed when text is formed with standard character fonts that cannot be changed. In systems that allow for changes in character size, string definitions can be windowed the same as other primitives. For characters formed with line segments, the mapping to the view port can be carried out as a sequence of line transformations.

From normalized coordinates, object descriptions are mapped to the various display devices. Any number of output devices can be open in a particular application, and another window-to-view port transformation can be performed for each open output device. This mapping, called the workstation transformation, IS accomplished by selecting a window area in normalized space and a view port area in the coordinates of the display device. With the workstation transformation, we gain some additional control over the positioning of parts of a scene on individual output devices. **As** illustrated in Fig.5, we can use workstation transformations to partition a view so that different parts of normalized space can be displayed on different output devices.



Mapping selected parts of a scene in normalized coordinates to different video monitors with workstation transformations.

Clipping Operation

Generally, any procedure that identifies those portions of a picture that are either inside or outside of a specified region of space is referred to as a clipping algorithm, or simply clipping. The region against which an object is to clipped is called a clip window.

Applications of clipping include extracting part of a defined scene for viewing; identifying visible surfaces in three-dimensiona1 views; anti aliasing line *seg*ments or object boundaries; creating objects using solid-modeling procedures;
Displaying a multi window environment; and drawing and painting operations that allow parts of a picture to be selected for copying, moving, erasing, or duplicating. Depending on the application, the clip window can be a general polygon or it can even have curved boundaries. We first consider clipping methods using rectangular clip regions, then we discuss methods for other clip region shapes.

In the following sections, we consider algorithms for clipping the following primitive types

- Point Clipping
- Line Clipping (straight-line segments)
- Area Clipping (polygons)
- Curve Clipping
- Text Clipping

Line and polygon clipping routines are standard components of graphics packages, but many packages accommodate curved objects, particularly spline curves and conics, such as circles and ellipses. Another way to handle curved objects is to approximate them with straight-line segments and apply the line or polygon clipping procedure.

Point clipping

Assuming that the clip window is a rectangle in standard position, we save a point P = ***(x,* y)** for display if the following inequalities are satisfied:

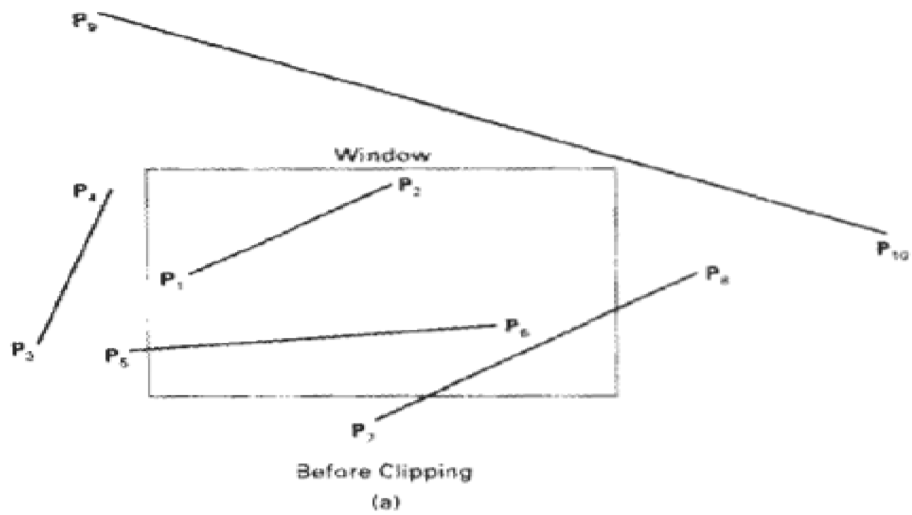
$$xw_{min} \leq x \leq xw_{max}$$

$$yw_{min} \leq y \leq yw_{max}$$

where the edges of the clip window $(xw_{min}, xw_{max}, yw_{min}, yw_{max})$ can be either the world-coordinate window boundaries or view port boundaries. If any one of these four inequalities is not satisfied, the point is clipped (not saved for display). Although point clipping is applied less often than line or polygon clipping, some .applications may require a point clipping procedure. For example, point clipping can be applied to scenes involving explosions or sea foam that are modeled with particles (points) distributed in some region of the scene.

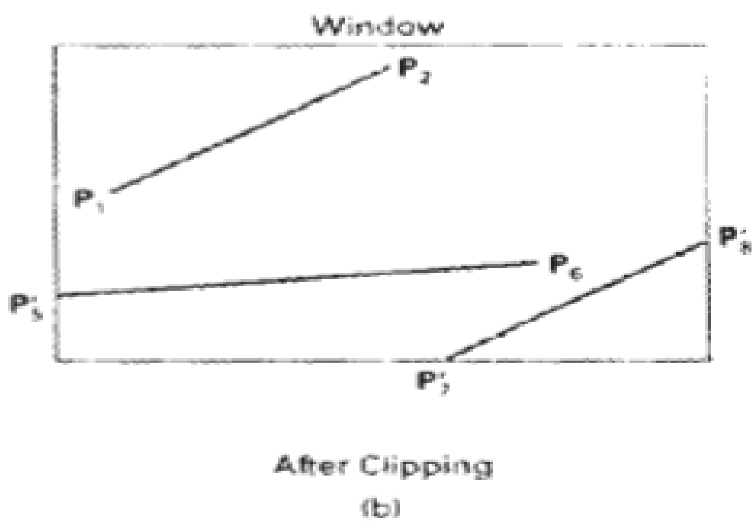
Line clipping

The relationships between line positions and a standard rectangular clipping region. A line clipping procedure involves several parts. First, we can test a given line segment to determine whether it lies completely inside the clipping window. If it does not, we try to determine whether it lies completely outside the window. Finally, if we cannot identify a line as completely inside or completely outside, we must perform intersection calculations with one or more clipping boundaries. We process lines through the "inside-outside" tests by checking the line endpoints. A line with both endpoints inside all clipping boundaries, such as the line from $\mathbf{P}_1$ to $\mathbf{P}_2$, is saved. A line with both endpoints outside any one of the clip boundaries (line $\overline{\mathbf{P}_3\mathbf{P}_4}$ in Figure) is outside the window.



Line clipping against a rectangular shape window.

All other lines cross one or more clipping boundaries, and may require calculation of multiple intersection points. to minimize calculations, we try to devise clipping algorithms that can efficiently identify outside lines and reduce intersection calculations.



For a line segment with endpoints $(x_1,y_1)$and(x_2,y_2) and one or both endpoints outside the clipping rectangle, the parametric representation.

$$X = x_v + u(x_2 - x_1)$$
$$Y = y_v + u(y_2 - y_1) \qquad 0<u<1$$

Could be used to determine values of parameter **u** for intersections with the clipping boundary coordinates. If the value of *u* for an intersection with a rectangle boundary edge is outside the range *0* to 1, the line does not enter the interior of the window at that boundary. If the value of *u* is with in the range from 0 to 1, the line segment does indeed cross into the clipping area. This method can be applied to each clipping boundary edge in turn to determine whether any part of the line segment is to be displayed. Line segments that are parallel to window edges can be handled as specia1 cases.

Clipping line segments with these parametric tests requires a good deal of Computation and faster approaches to clipping are possible. A number of efficient line clippers have been developed, and we survey the major algorithms in the next section. Some algorithm are designed explicitly for two-dimensional Pictures and some are easily adapted to three-dimensional applications.

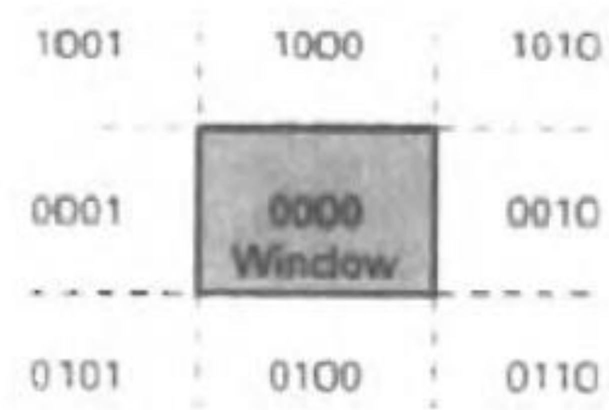
Cohen-Sutherland line clipping

This is one of the oldest and most popular line-clipping procedures. Generally, the method speeds up the processing of line segments performing initial tests that reduce the number of intersections that must he calculated. Every line end Point in a picture is assigned a four-digit binary code, called a region code that identifies the location of the point relative to the boundaries of the clipping rectangle. Regions are set up in reference to the boundaries as shown in Fig.7**.** Each bit position in the region code is used to indicate one of the four relative coordinate positions of the point with respect to the clip window: to the left, right, top, or bottom. By numbering the bit positions in the region code as 1 through 4 from right to left, the coordinate regions can be correlated with the bit positions as bit 1: left

bit 2: right

bit 3: below

bit 4: above

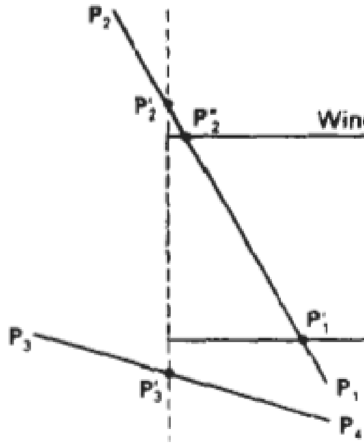


Binary region codes assigned to line endpoints according to relative position with respect to the clipping rectangle.

A value of 1 in any bit position indicates that the point is in that relative position; Otherwise, the bit position is set to 0. If a point is within the clipping rectangle, the region code is 0000. A point that is below and to the left of the rectangle has a region code of 0101. Bit values in the region code are determined by comparing endpoint Coordinate values (x, y) to the clip boundaries. Bit 1 is set to 1 if x $< xw_{min}$,. The other three bit values can be determined using similar comparisons. For languages in which bit manipulation is possible, region-code bit values can be determined with the following two steps: (1) Calculate differences between endpoint coordinates and clipping boundaries. (2) Use the resultant sign bit of each difference calculation to set the corresponding value in the region code. Bit **1** is the sign bit of ***x*** - bit 2 is the sign bit of $xw_{max} - x$; bit 3 is the sign bit of $y - yw_{min}$; and bit 4 is the sign bit of $yw_{max} - y$.

Once we have established region codes for all line endpoints, we can quickly determine which lines are completely inside the clip window and which are clearly outside. Any lines that are completely contained within the window boundaries have a region code of 0000 for both endpoints, and we trivially accept these lines. Any lines that have a 1 in the same bit position in the region codes for each endpoint are completely outside the clipping rectangle, and we trivially reject these lines. We would discard the line that has a region code of 1001 for one endpoint and a code of 0101 for the other endpoint. Both endpoints of this line are left of the clipping rectangle, as indicated by the 1 in the first bit position of each region code. A method that can be used to test lines for total clipping is to perform the logical and operation with both region codes. If the result is not 0000, the line is completely outside the clipping region.

Lines that cannot be identified as completely inside or completely outside a Clip window by these tests are checked for intersection with the window boundaries. As shown in Fig.8, such lines may or may not cross into the window interior. We begin the clipping process for a line by comparing an outside endpoint to a clipping boundary to determine how much of the line can be discarded. Then the remaining part of the Line is checked against the other boundaries, and we continue until either the line is totally discarded or a section is found inside the window. We set up our algorithm to check line endpoints against clipping boundaries in the order left, right, bottom, top.



Lines extending from one coordinate region to another may pass through the clip window, or they may intersect clipping boundaries without entering the window.

To illustrate the specific steps in clipping lines against rectangular boundaries using the Cohen-Sutherland algorithm, we show how the lines in Figure could be processed. Starting with the bottom endpoint of the line from P_1 to P_2

We check P, against the left, right, and bottom boundaries in turn and find that this point is below the clipping rectangle. We then find the intersection point Pi with the bottom boundary and discard the line section from P_1 to P'_1. The line now has been reduced to the section from P'_1 to P_2 since P_2 is outside the clip window, we check this endpoint against the boundaries and find that it is to the left of the window. Intersection point P'_2 is calculated, but this point is above the window. So the final intersection calculation yields P''_2, and the line from P'_1 to P''_2 is saved. This completes processing for this line, so we save this part and go on to the next line. Point P_3 in the next line is to the left of the clipping rectangle, so we determine the intersection P'_3 and eliminate the line section from P_3 to P'_3. By checking region codes for the line section from P'_3 to P_4, we find that the remainder of the line is below the clip window and can be discarded also.

Intersection points with a clipping boundary can be calculated using the

Slope-intercept form of the line equation. For a line with endpoint coordinates (x,y,) and (x2, y2), they coordinate of the intersection point with a vertical boundary can be obtained with the calculation.

$$Y=y_1+m(x-x_1)$$

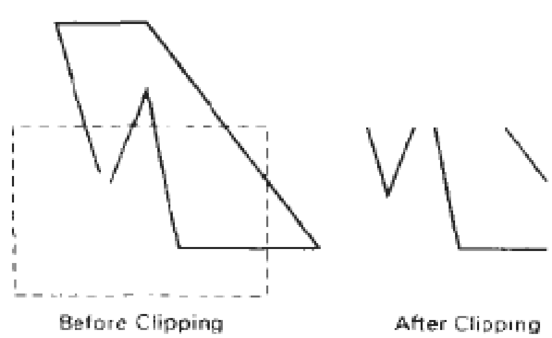
where the ***x*** value is set either to xw_{min} to xw_{max} and the slope of the line is calculated as m=(y2-y1)/(x2-x1).Similarly, if we are looking for the intersection with a horizontal boundary, the ***x*** coordinate can be calculated as

$$x = x_1 + \frac{y - y_1}{m}$$

with y set either to yw_{min} or to yw_{max}

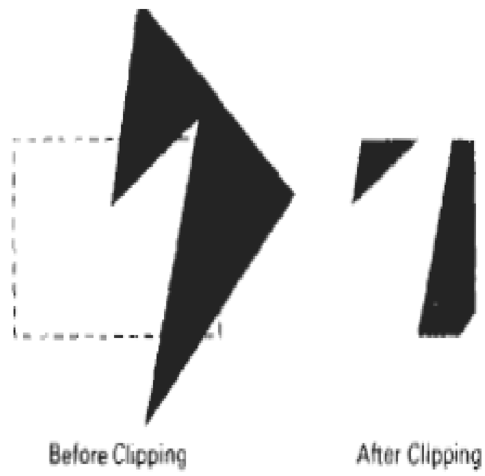
Polygon clipping

To clip polygons, we need to modify the line-clipping procedures discussed in the previous section. A polygon boundary processed with a line clipper may be displayed as a series of unconnected line segments Figure depending on the Orientation of the polygon to the clipping window.



Display of a polygon processed by a line-dipping algorithm

What we really want to display is a bounded area after clipping, as in Figure. For polygon clipping, we require an algorithm that will generate one or more closed areas that are then scan converted for the appropriate area fill. The output of a polygon clipper should be a sequence of vertices that defines the clipped polygon boundaries.

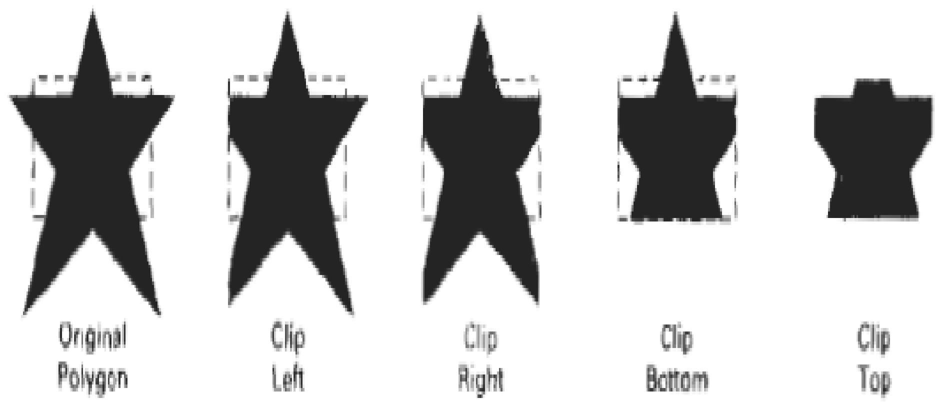


Display of a correctly clipped polygon.

Sutherland-Hodgeman polygon clipping

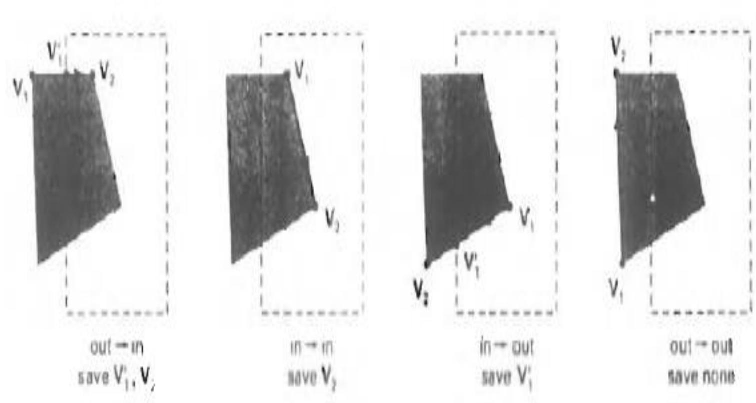
We can correctly clip a polygon by processing the polygon boundary as a whole against each window edge. This could be accomplished by processing all polygon vertices against each clip rectangle boundary in turn. Beginning with the initial set of polygon vertices, we could first clip the polygon against the left rectangle boundary to produce a new sequence of vertices. The new set of vertices could then k successively passed to a right boundary clipper, a bottom boundary clipper, and a top boundary clipper, as in

At each step, a new sequence of output vertices is generated and passed to the next window boundary clipper.



Clipping a polygon against successive window boundaries.

There are four possible cases when processing vertices in sequence around the perimeter of a polygon. As each pair of adjacent polygon vertices is passed to a window boundary clipper, we make the following tests: (1) If the first vertex is Outside the window boundary and the second vertex is inside, both the intersection point of the polygon edge with the window boundary and the second vertex are added to the output vertex list. *(2)* If both input vertices are inside the window boundary, only the second vertex is added to the output vertex list. (3) if the first vertex is inside the window boundary and the second vertex is outside, only the edge intersection with the window boundary is added to the output vertex list. (4) If both input vertices are outside the window boundary, nothing is added to the output list. These four cases are illustrated in Figure for successive pairs of polygon vertices. Once all vertices have been processed for one clip window boundary; the output 1ist of vertices is clipped against the next window boundary.

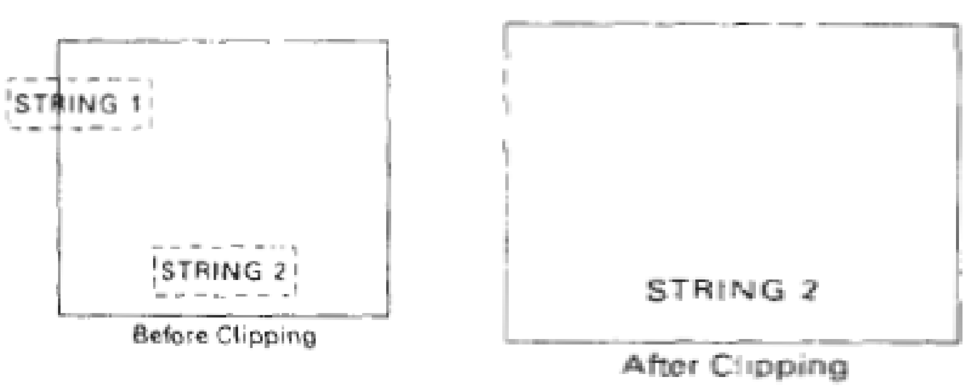


Successive processing of pairs of polygon vertices against the left window boundary.

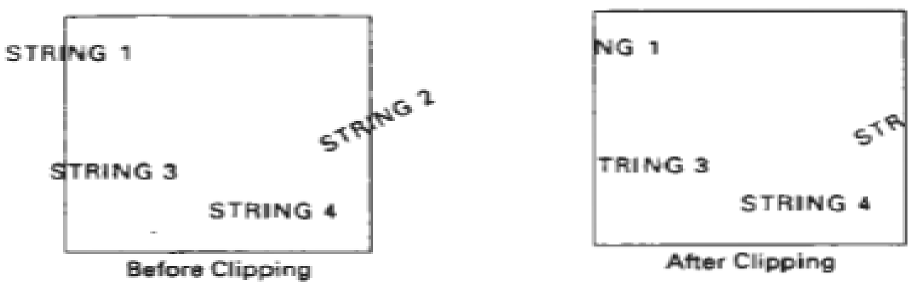
Text clipping

There are several techniques that can be used to provide text clipping in a graphics package. The clipping technique used will depend on the methods used to generate characters and the requirements of a particular application.

The simplest method for processing character strings relative to a window boundary is to use the all-or-none string-clipping strategy shown in Fig.13. If all of the string is inside a clip window, we keep it. Otherwise, the string is discarded. This procedure is implemented by considering a bounding rectangle around the text pattern. The boundary positions of the rectangle are then compared to the window boundaries, and the string is rejected if there is any overlap. This method produces the fastest text clipping.



An alternative to rejecting an entire character string that overlaps a window boundary is to use the all-or-none character-clipping strategy. Here we discard only those characters that are not completely inside the window .In this case, the boundary limits of individual characters are compared to the window. Any character that either overlaps or is outside a window boundary is clipped.



Text clipping using a bounding &tangle about individual characters

A final method for handling text clipping is to clip the components of individual Characters. We now treat characters in much the same way that we treated lines. If an individual character overlaps a clip window boundary, we clip off the Parts of the character that are outside the window Outline character Fonts formed with line segments can be processed in this way using a line clipping algorithm. Characters defined with bit maps would be clipped by comparing the relative position of the individual pixels in the character grid patterns to the clipping boundaries.

POSSIBLE QUESTIONS
UNIT – IV

Part-A

Online Examinations (One marks)

1. Screen locations are referred with ______
   a) Parallel lines b) Digital Differential Analyzer
   c) **integer value** d) parameters
2. To load an intensity value into the frame buffer at a position corresponding to column x along scan line y, we have available a__________ of the form
   a) **Low level procedure** b) High level procedures
   c) Cartesian Slope d) Deflection slope
3. The Cartesian slope-intercept equation for a straight line is _____
   a)Y or y + 1 b) y=m,y+5 c)x+5 d)**y=m.x+6**
4. On raster systems lines are plotted with ______
   a) **Pixels** b) points c)lines d)parallel
5. For lines with slope magnitude|<1,Δx can be set proportional to a small
   a) Vertical deflection b) positive slope
   c) negative slope d) **horizontal deflection voltage**
6. _________ Projection preserves the shape and size of the projected object.
   a)**Parallel** b) Perspective c)both a and b d)None of the above
7. _________ algorithm does not suit for 3 dimensional clipping.
   a)Cohen Sutherland algorithm b) **Bresenham algorithm**
   c)Sutherland Hodgman algorithm d)None of the above
8. _________ defines the spatial extent that is visible through the rectangular window of the view plane.
   a)view point b) **view volume** c)viewport d)view region
9. The receptor cells in the retina of the eye, that are sensitive to color is _________
   a) Rods b) **cones** c) cubes d) lines
10. The receptor cells in the retina of the eye, that are sensitive to color is _________
    a) **Rods** b) cones c) cubes d) lines

Part B

Essay Type (8 Marks)

1. Distinguish between window port and view port?
2. List out the Clipping Operation.
3. What is fixed point scaling?
4. List out the Clipping Operation.
5. What is Texture?
6. Write short notes on Text Clipping.
7. Explain the method of transforming the co-ordinates from window to View port.
8. Explain in detail about 2 clipping operations.
9. Discuss about Point Clipping and Line clipping.
10. Discuss about Cohen Sutherland Line Clipping
11. Discuss about curve Clipping and Text clipping.
12. Explain in detail about polygon clipping operations.

KARPAGAM ACADEMY OF HIGHER EDUCATION

COIMBATORE - 21

DEPARTMENT OF COMPUTER SCIENCE

CLASS : III B.Sc COMPUTER SCIENCE

BATCH : 2015-2018

Part -A Online Examinations **(1 mark questions)**

SUBJECT: COMPUTER GRAPHICS **SUBJECT CODE: 15CSU503**

UNIT-IV

S.No	Questions	option1	option2	option3	option4	Answer
1	_________ is defined as a mapping of a point P to its image P' in the view plane.	Transformation	Viewing	Clipping	Projection	Projection
2	The mapping is determined by a line that passes through point P and intersects the view plane, this line	projector	normal	vetor	axes	projector
3	Which of these are basic types of projections	parallel projection	perspective projection	Both	None of the above	Both
4	________ projection is called converging projectors.	Parallel	Perspective	Isometric	Axonometric	Perspective
5	__________ drawings are characterised by vanishing points.	Parallel	Perspective	Isometric	Axonometric	Perspective
6	The perspective projection gives the illusion that certain set of parallel lines appear to meet at a point called	view point	projection point	vanishing point	reference point	vanishing point
7	_________ projection preserves the shape and size of the projected object.	Parallel	Perspective	both	None of the above	Parallel
8	_________ algorithm does not suit for 3 dimensional clipping.	Cohen Sutherland	Bresenham algorithm	Sutherland Hodgman	None of the above	Bresenham algorithm

9	_________ defines the spatial extent that is visible through the rectangular window of the view plane.	view point	view volume	viewport	view region	view volume
10	The view plane is also called ___________	projection plan	view direction	mapping funtion	reference vector	projection plan
11	The Center of projection is called the ____	view point	view volume	viewport	reference vector	view point
12	Three dimensional viewing of objects requires the specification of _________	view plane	view point	view volume	All the Above	All the Above
13	The unit normal vector N = n1I + n2J + n3 K , |N| = __________	0	1	2	3	1
14	N = n1I + n2J + n3 K in 3D viewing is called the _____	Reference vector	Translation Vector	Unit Normal vector	Up vector	Unit Normal vector
15	Reference Vector U is also known as _________	Co-ordinate Vector	Translation Vector	Unit Normal vector	Up vector	Up vector
16	The triad formed by vectors Ip, Jp and N is called _____	Cartesian co-ordinate system	Vector Coordinate	Viewing coordinate	None of the above	Viewing coordinate system
17	______ coordinate system is choosen because the p and q coordinate axes are superimposed on the display	Left handed	Right handed	Cartesian	3 dimensional	Left handed
18	In Left handed coordinate system the normal vector N point _________,facing the display.	towards the viewer	away from the viewer	the right side of the viewer	left side of the viewer	away from the viewer
19	In Left handed coordinate system increasing distance away from the observer is measured along _________	Ip	Jp	N	Up	N
20	If the xy plan is the view plane then __________ measures the depth or distance of the point from the	x coordinate	y coordinate	z coordinate	None of the above	z coordinate
21	If the view plan is the xy plane then Ip = I, Jp = J and N = __________	Np	-K	-N	z	-K

22	The right handed world coordinates (x,y,z) can be changed to left handed view plane coordinates (x',y',z') by performing the transformation where x'=x,y'=y and z'	z	-z	k	-k	-z
23	The ________ bounds a region in world coordinate space that is clipped and projected to the view plan.	view point	view volume	viewport	view region	view volume
24	The region bounded by the view volume is_________ to the viewplane.	transformed	mapped	clipped	clipped and projected	clipped and projected
25	For a __________ , the view volume corresponding to the given window is a semi-infinite pyramid.	parallel view	prespective view	Axanometric view	Orthographic view	prespective view
26	For parallel projected views, the view volume is ____	a semi infinite pyramid	finite parallelogram	infinite parallelopiped	finite polygon	infinite parallelopiped
27	The view volumes are ______ in extent	finite	infinite	invisible	fixed	infinite
28	For perspective views, very distant objects appear as ________	a semi infinite pyramid	infinite parallelopiped	disjointed structure	indistinguishable spots	indistinguishable spots
29	For perspective views, objects very close to the center of projection appear as ________	disjointed structure	indistinguishable spots	infinite parallelopiped	finite polygon	disjointed structure
30	A finite volume is delimited by front and back clipping planes ________ to the view plane	perpendicular	parallel	similar	normal	parallel
31	In _________ clipping strategy, clipping is done against the original view volume	canonical	normalized	tranformal	direct	direct
32	In _________ clipping strategy, a normalizing transformation is applied to the original view volume	canonical	normalized	tranformal	direct	canonical
33	When normalizing transformation is applied to the original view volume it is called _________ view	canonical	normalized	tranformal	direct	canonical
34	The canonical view volume for a parallel projection is the ________	infinte pyramid	infinte parallelopiped	unit cube	truncated pyramid	unit cube

35	The canonical view volume for a perspectivel projection is the ________	infinte pyramid	infinte parallelopiped	unit cube	truncated pyramid	truncated pyramid
36	After clipping in viewing coordinates, the resulting structure is projected onto the ________	clipping window	projector	World Coordinate	screen projection plane	screen projection plane
37	In 3D graphics pipeline, viewing transformation and projection are carried out according to the _____	viewing	projecting	transforming	colour	viewing
38	In 3D graphics pipeline, the result of projection is mapped to workstation viewport via ________	3D viewing transformation	2D viewing transformation	scaling transformation	Rotation transformation	2D viewing transformation
39	Which of these is a complex geometric form.	point	line	curves	polygon	curves
40	_________ and _________ are the basic building blocks of computer graphics.	points and lines	lines and curves	points and curves	polylines and curves surface	points and lines
41	________ is a chain of connected line segments.	points	curves	quadric surfaces	polylines	quadric surfaces
42	__________ is a closed polyline	points	curves	quadric surfaces	polygon	polygon
43	A _________ is a polygon in which all vertices lie on the same plane	convex polygon	concave polygon	planar polygon	None of the above	planar polygon
44	The _________is also called polygonal net or polygonal mesh.	planar polygon	wireframe model	polyline	quadric surface	wireframe model
45	A _________ is a closed polygonal net in which each polygon is planar.	curved surface	polyline	polyhedron	quadric surface	polyhedron
46	__________ is a method of representing a polygonal net model.	Explicit vertex listing	Explicit edge listing	Polygon listing	All the Above	All the Above
47	The polygons are called the __________ of the polyhedron	edges	vertices	faces	sides	faces
48	Which of these is complex in wire frame models	constructing wireframe	representing curved	applying geometic	applying coordinate	representing curved surfaces

49	A model constructed using solid objects as building blocks is called ______	Wireframe modeling	Solid object modeling	Block modeling	None of the above	Solid object modeling
50	________ is a process of building model by assembling simpler objects.	Additive modeling	Subtractive Modeling	Destructive modeling	Constructive modeling	Additive modeling
51	________ is a process of removing pieces from a given object to create a new object.	Additive modeling	Subtractive Modeling	Destructive modeling	Constructive modeling	Subtractive Modeling
52	A curve is specified by __________	polylines	curved surface pathes	a set of control points	two endpoints	a set of control points
53	The Control points control the __________ of the curve.	length	shape	slope	width	shape
54	Curves are generally __________ with respect to any coordinate system.	single valued	bi-valued	multi-valued	None of the above	None of the above
55	If the movement of the control point affects the shape of the curve only in a small neighbourhood of the control	Coordinate independence	Variation diminishing	Versatility	Local controllability.	Local controllability.

KARPAGAM ACADEMY OF HIGHER EDUCATION

(Deemed University Established Under Section 3 of UGC Act 1956)
Coimbatore - 641021.
(For the candidates admitted from 2015 onwards)
DEPARTMENT OF COMPUTER SCIENCE, CA & IT

SUBJECT : COMPUTER GRAPHICS
SEMESTER : V
SUBJECT CODE: 15CSU503 CLASS : III B.Sc.CS

UNIT-V
Three – Dimensional Display Methods-Parallel Projection- three Dimensional Geometric Transformations: Translation-Rotations- Scaling. **Projections**: Parallel Projections-Perspective Projections. **Visible Surface Detection Methods**: Classification of Visible Surface Detection Algorithms-Back Face Detection-Depth Buffer Method-Depth Buffer Method- Area Sub division Method.

TEXT BOOK

1. Donald Hearn and M. Pauline Baker. 2007. *Computer Graphics-C Version*, 2nd Edition, Pearson Education, New Delhi.
   (Page Nos. : 24-54, 56-77, 80-92, 103-118, 204-215, 236-256, 427-443, 458-463, 490-495, 502-505)

REFERENCES

1. Amarendra N. Sinha. 2008. *Computer Graphics*, 1st Edition ,Tata McGraw Hill, New Delhi.
2. Foley, Vandam, Feiner and Hughes. 1999. *Computer Graphics Principles and Practices*, 2nd Edition ,Addison Wesley, Singapore.
3. Evangeline D., Anita, S. 2016. *Computer Graphics And Multimedia Insights, Mathematical Models And Programming Paradigms*. PHI Learning Pvt. Ltd.
4. Rae Earnshaw.2014.*Computer Graphics: Developments in Virtual Environments*, 2nd Edition ,academic Press, New Delhi.

WEB SITES:
www.cgshelf.com
www.cgtutorials.com
www.allgraphicdesign.com

UNIT-5

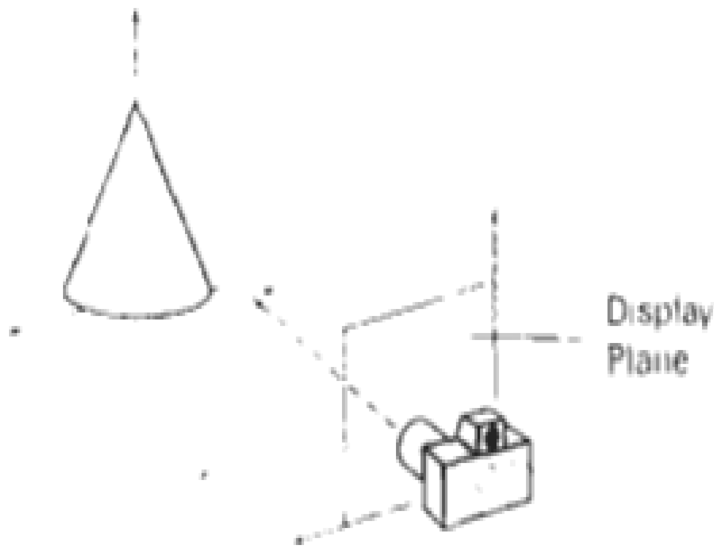
Introduction

When we model and display a three-dimensional scene, there are many more considerations we must take into account besides just including coordinate values for the third dimension. Object boundaries can be constructed with various combinations of plane and curved surfaces, and we sometimes specify information about object interiors. Graphics packages often provide routines for displaying internal components or cross-sectional views of solid objects. Also, some geometric transformations are more involved in three-dimensional space than in two dimensions.

For example, we can rotate an object about an axis with any spatial orientation in three-dimensional space. Two-dimensional rotations, on the other hand, are always around an axis that is perpendicular to the xy plane. Viewing transformations in three dimensions are much more complicated because we have many more parameters to select when specifying how a three-dimensional scene is to be mapped to a display device. The scene description must be processed through viewing-coordinate transformations and projection routines that transform three-dimensional viewing coordinates onto two-dimensional device coordinates. Visible parts of a scene; for a selected view, must he identified; and surface-rendering algorithms must he applied if a realistic rendering of the scene is required.

Three dimensional display methods

To obtain a display of a three-dimensional scene that has been modeled in world coordinates. We must first set up a coordinate reference for the "camera". This coordinate reference defines the position and orientation for the plane on the camera film Figure, which is the plane we want to use to display a view of the objects in the scene. Object descriptions are then transferred to the camera reference coordinates and projected onto the selected display plane. We can then display the objects in wire frame (outline) form; we can apply lighting and surface rendering techniques to shade the visible surfaces.



Coordinate reference for obtaining a particular view of a three dimensional scene

Three Dimensional Geometric Transformations

Methods for geometric transformations and object modeling in three dimensions are extended from two-dimensional methods by including considerations for the ***z*** coordinate. We now translate an object by specifying a three-dimensional translation vector, which determines how much the object is to be moved in each of the three coordinate directions. Similarly, we scale an object with three coordinate scaling factors. The extension for three-dimensional rotation is less straightforward. When we discussed two-dimensional rotations in the *xy* plane, we needed to consider only rotations about axes that were perpendicular to the *xy* plane. In three-dimensional space, we can now select any spatial orientation for the rotation axis. Most graphics packages handle three-dimensional rotation as a composite of three rotations, one for each of the three Cartesian axes. Alternatively, a user can easily set up a general rotation matrix, given the orientation of the axis and the quired rotation angle. As in the two-dimensional case, we express geometric transformations in matrix form. Any sequence of transformations is then represented as, a single matrix, formed by concatenating the matrices for the individual transformations in the sequence.

Translation

In a three-dimensional homogeneous coordinate representation, a point is translated from position P = *(x,* y, *z)* to position P' = *(x'*, y', z') with the matrix operation

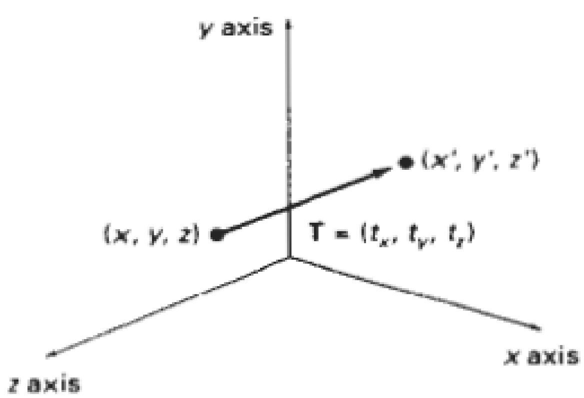
$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Or

$$\mathbf{P'} = \mathbf{T} \cdot \mathbf{P}$$

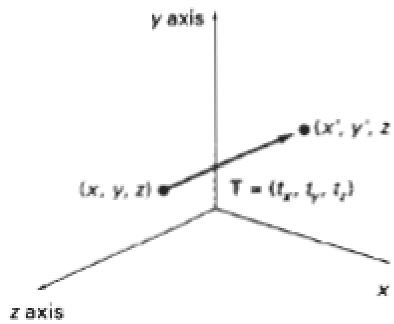
Parameters $t_x, t_y, and\ t_z$, specifying translation distances for the coordinate directions ***x*, y,** and z, are assigned any real values. The matrix representation is equivalent to the three equations,

$$x' = x + t_x, \qquad y' = y + t_y, \qquad z' = z + t_z$$



Translating a point with translation vector $\mathbf{T} = (t_x, t_y, t_z)$.

An object is translated in three dimensions by transforming each of the defining points of the object. For an object represented as a set of polygon surfaces, we translate each vertex of each surface Figure and redraw the polygon facets in the new position.



Translating an object with translation vector T.

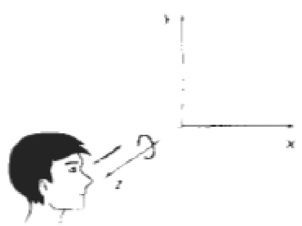
We obtain the inverse of the translation matrix in Eq. 11-1 by negating the translation distances t_x,t_y and t_z this produces a translation in the opposite direction, and the product of a translation matrix and its inverse produces the identity Matrix.

Rotation

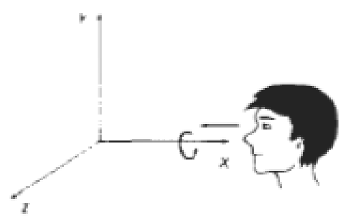
To generate a rotation transformation for an object, we must designate an axis of rotation (about which the object is to be rotated) and the amount of angular rotation. Unlike two-dimensional applications, where all transformations are carried out in the xy plane, a three-dimensional rotation can be specified around any line in space. The easiest rotation axes to handle are those that are parallel to the coordinate axes. Also, we can use combinations of coordinate axis rotations (along with appropriate translations) to specify any general rotation.

By convention, positive rotation angles produce counterclockwise rotations about a coordinate axis, if we are looking along the positive half of the axis toward the coordinate origin Figure. This agrees with our earlier discussion of Rotation in two dimensions, where positive rotations in the xy plane are counterclockwise about axes parallel to the z axis.

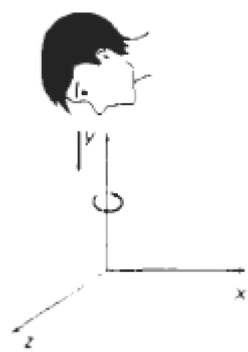
Fig3: Positive rotation directions about the coordinate axes are Counterclockwise, when looking toward the origin from a positive coordinate position on each axis.



(a)



(b)



(c)

Coordinate-Axes Rotations

The two-dimensional z-axis rotation equations are easily extended to three dimensions:

$$x' = x\cos\theta - y\sin\theta$$

$$y' = x\sin\theta + y\cos\theta$$

$$z' = z$$

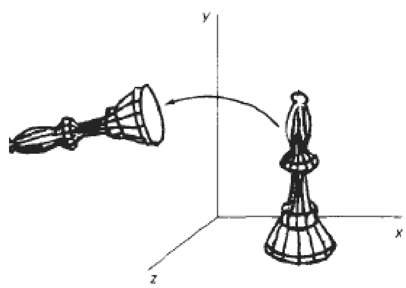
Parameter θ specifies the rotation angle. In homogeneous coordinate form, the three-dimensional z-axis rotation equations are expressed as,

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

which we can write more compactly as,

$$\mathbf{P}' = \mathbf{R}_z(\theta) \cdot \mathbf{P}$$

Figure illustrates rotation of an object about the z axis.



Rotation of an object about the z axis

Transformation equations for rotations about the other two coordinate axes can be obtained with a cyclic permutation of the coordinate parameters ***x*,** y, and z in Eqs. 11-4. That is, we use the replacements

$$x \rightarrow y \rightarrow z \rightarrow x$$

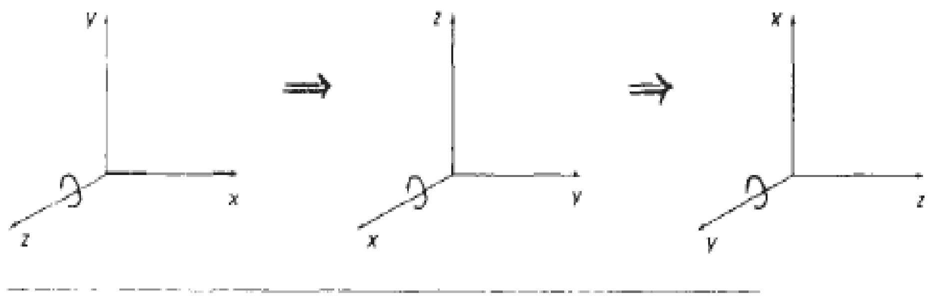
as illustrated in Figure.

Substituting permutations, we get the equations for an x-axis rotation:

$$y' = y\cos\theta - z\sin\theta$$
$$z' = y\sin\theta + z\cos\theta$$
$$x' = x$$

which can be written in the homogeneous coordinate form

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta & 0 \\ 0 & \sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$



Cyclic permutation of the Cartesian-coordinate axes to produce the three sets of coordinate-axis rotation equations.

Cyclically permuting coordinates give us the transformation equations for a y-axis rotation:

$$z' = z\cos\theta - x\sin\theta$$
$$x' = z\sin\theta + x\cos\theta$$
$$y' = y$$

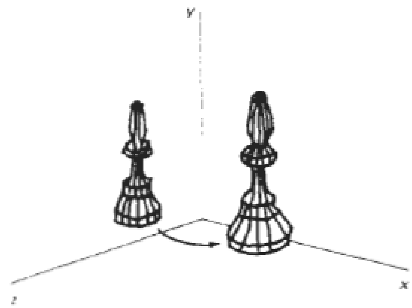
The matrix representation for y-axis rotation is

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & 0 & \sin\theta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\theta & 0 & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Or

$$\mathbf{P}' = \mathbf{R}_y(\theta) \cdot \mathbf{P}$$

An example of y-axis rotation is shown in Figure



Rotation of an object about the y axis.

An inverse rotation matrix is formed by replacing the rotation angle 0 by $-\theta$. Negative values for rotation angles generate rotations in a clockwise direction, so the identity matrix is produced when any rotation matrix is multiplied by its inverse. Since only the sine function is affected by the change in sign of the rotation angle, the inverse matrix can also be obtained by interchanging rows and columns. That is, we can calculate the inverse of any rotation matrix R by evaluating its transpose $(\mathbf{R}^{-1} = \mathbf{R}^T)$. This method for obtaining an inverse matrix holds also for any composite rotation matrix.

General Three-Dimensional Rotations

A rotation matrix for any axis that does not coincide with a coordinate axis can be set up as a composite transformation involving combinations of translations and the coordinate-axes rotations. We obtain the required composite matrix by first setting up the transformation sequence that moves the selected rotation axis onto one of the coordinate axes. Then we set up the rotation matrix about that coordinate axis for the specified rotation angle. The last step is to obtain the inverse transformation sequence that returns the rotation axis to its original position.

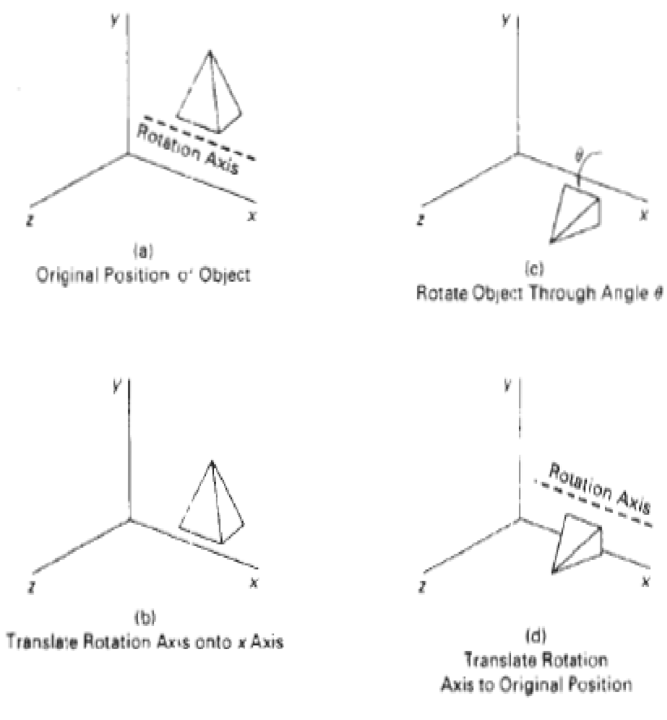
In the special case where an object is to be rotated about an axis that is parallel to one of the coordinate axes, we can attain that
desired rotation with the following transformation sequence.

1. Translate the object so that the rotation axis coincides with the parallel coordinate axis.
2. Perform the specified rotation about that axis.

3. Translate the object so that the rotation axis is moved back to its original position.

The steps in this sequence are illustrated in Figure. Any coordinate position P on the object in this figure is transformed with the sequence shown as,

$$P' = T^{-1} \cdot R_x(\theta) \cdot T \cdot P$$



Sequence of transformations for rotating an object about an axis that is parallel to the *x* axis

where the composite matrix for the transformation is,

$$R(\theta) = T^{-1} \cdot R_x(\theta) \cdot T$$

which is of the same form as the two-dimensional transformation sequence for rotation about an arbitrary pivot point.

When an object is to be rotated about an axis that **is** not parallel to one of the coordinate axes, we need to perform some additional transformations. In this case, w also need rotations lo align the axis with a selected coordinate axis and to bring the axis hack to its original orientation. Given the specifications for the rotation axis and the rotation angle, we can accomplish the required rotation in five steps.

1 Translate the object so that the rotation axis pass= through the coordinate origin.

2. Rotate the object so that the axis of rotation coincides with one of the coordinate axes.

3. Perform the specified rotation about that coordinate axis.

4. Apply inverse rotations to bring the rotation axis back to its
original orientation.

5. Apply the inverse translation to bring the rotahon axis back to its original position.

Scaling

The matrix expression for the scaling transformation of a position P = (x, y, *z)* relative to the coordinate origin can be written as

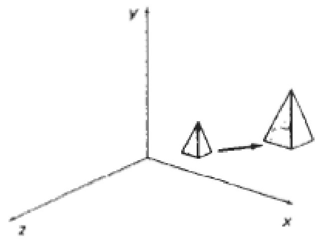
$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Or

$$\mathbf{P}' = \mathbf{S} \cdot \mathbf{P}$$

Where scaling parameters $s_{x,\ syandszare}$ assigned any positive values. Explicit expressions for the coordinate transformations for scaling relative to the origin are

$$x' = x \cdot s_x, \qquad y' = y \cdot s_y, \qquad z' = z \cdot s_z$$



Doubling the sue of an object with transformation 11-42 also moves the object farther from the origin

Scaling an object with transformation 11-42 changes the size of the object and repositions the object relative to the coordinate origin. Also, if the transformation parameters are not all equal, relative dimensions in the object are changed: We preserve the original shape of an object with a uniform scaling $(s_x = s_y = s_z)$. The result of scaling an object uniformly with each scaling parameter set to 2 is shown in Figure.

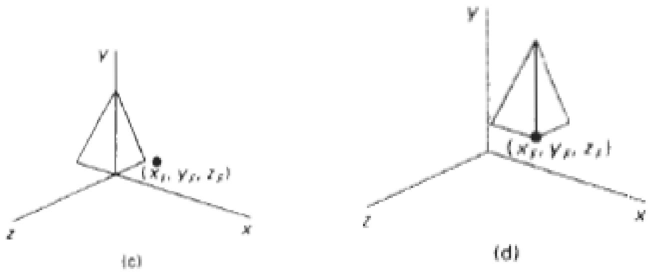
Scaling with respect to a selected fixed position (x_f, y_f, z_f) can be represented with the following transformation sequence:

1. Translate the fixed point to the origin.
2. Scale the object relative to the coordinate origin
3. Translate the fixed point back to its original position.

This sequence of transformations is demonstrated in Figure. The matrix representation for an arbitrary fixed-point scaling can then be expressed as the concatenation of these translate-scale-translate transformations as

$$\mathbf{T}(x_f, y_f, z_f) \cdot \mathbf{S}(s_x, s_y, s_z) \cdot \mathbf{T}(-x_f, -y_f, -z_f) = \begin{bmatrix} s_x & 0 & 0 & (1-s_x)x_f \\ 0 & s_y & 0 & (1-s_y)y_f \\ 0 & 0 & s_z & (1-s_z)z_f \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

(a) (b)



Scaling an object relative to the selected fixed point is equivalent to the sequence of transformations shown.

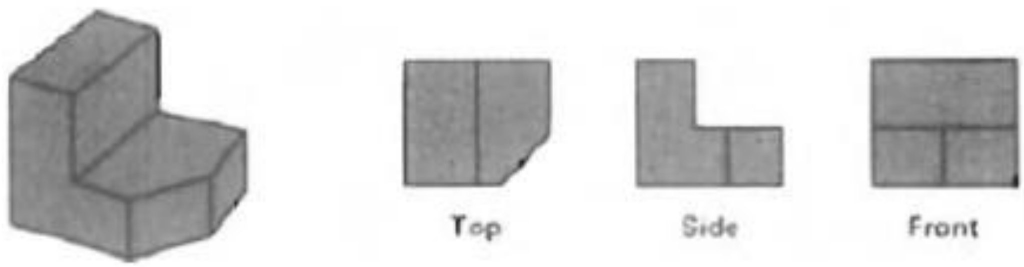
We form the inverse scaling matrix for either by replacing the scaling parameters s_x, s_y and s_z with their reciprocals. The inverse matrix generates an opposite scaling transformation, so the concatenation of any scaling matrix and its inverse produces the identity matrix.

Projections

Parallel Projection

One method for generating a view of a solid object is to project points on the object surface along parallel lines onto the display plane. By selecting different viewing positions, we can project visible points on the object onto the display plane to obtain different two-dimensional views of the object, as in Figure.

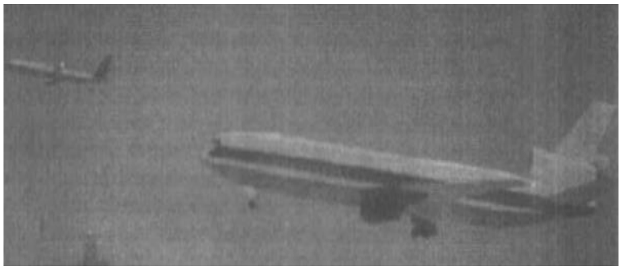
In a Parallel projection, parallel lines in the world-coordinate scene projected into parallel lines on the two-dimensional display plane. This technique is used in engineering and architectural drawings to represent an object with a set of views that maintain relative proportions of the object. The appearance of the solid object can then be reconstructed from the major views.



Three parallel-projection views of an object, showing relative proportions from different viewing positions.

Perspective Projection

Another method for generating a view of a 3 dimensional scene is to project points to the display plane along converging paths. This causes objects farther from the viewing position to be displayed smaller than objects of the same size that are nearer to the viewing position. In a perspective projection, parallel lines in a scene that are not parallel to the display plane are projected into converging lines. Scenes displayed using perspective projections appear more realistic, since this is the way that our eyes and a camera lens form images. In the perspective projection view shown in Figure, parallel lines appear to converge to a distant point in the background, and distant objects appear smaller than objects closer to the viewing position.



A perspective-projection view of an airport scene

Visible Surface Detection Methods

A major consideration in the generation of realistic graphics displays is identifying those parts of a scene that are visible from a chosen viewing position. There are many approaches we can take to solve this problem, and numerous algorithms have been devised for efficient identification of visible objects for different types of applications. Some methods require more memory, some involve more processing time, and some apply only to special types of objects. Deciding upon a method for a particular application can depend on such factors as the complexity of the scene, type of objects to be displayed, available equipment, and whether static or animated displays are to be generated. The various algorithms are referred to as visible-surface detection methods. Sometimes these methods are also referred to as hidden-surface elimination methods, although there can be subtle differences between identifying visible surfaces and eliminating hidden surfaces. For wire frame displays, for example, we may not want to actually eliminate the hidden surfaces, but rather to display them with dashed boundaries or in some other way to retain information about their shape. In this chapter, we explore some of the most commonly used methods for detecting visible surfaces in a three-dimensional scene.

Classification of Visible Surface Detection algorithm

Visible-surface detection algorithms are broadly classified according to whether they deal with object definitions directly or with their projected images. These two approaches are called object-space methods and image-space methods, respectively.

An object-space method compares objects and parts of objects to each other within the scene definition to determine which surfaces, as a whole, we should label as visible. In an image-space algorithm, visibility is decided point by point at each pixel position on the projection plane. Most visible-surface algorithms use image-space methods, although object space methods can be used effectively to locate visible surfaces in some cases. Line display algorithms, on the other hand, generally use object-space methods to identify visible lines in wire frame displays, but many image-space visible-surface algorithms can be adapted easily to visible-line detection.

Although there are major differences in the basic approach taken by the various visible-surface detection algorithms, most use sorting and coherence methods to improve performance. Sorting is used to facilitate depth comparisons by ordering the individual surfaces in a scene according to their distance from the view plane. Coherence methods are used to take advantage of regularities in a scene. An individual scan line can be expected to contain intervals (runs) of constant pixel intensities, and scan-line patterns often change little from one line to the next. Animation frames contain changes only in the vicinity of moving objects. And constant relationships often can be established between objects and surfaces in a scene.

Back Face Detection

A fast and simple object-space method for identifying the back faces of a polyhedron is based on the "inside-outside" tests discussed in Chapter 10. A point *(x*, y, z) is "inside" a polygon surface with plane parameters A, B, C, and D if

$$Ax + By + Cz + D < 0$$

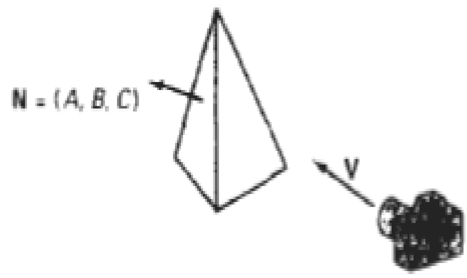
When an inside point is along the line of sight to the surface, the polygon must be a back face (we are inside that face and cannot see the front of it from our viewing position).

We can simplify this test by considering the normal vector N to a polygon surface, which has Cartesian components (A, B, C). In general, if V is a vector in the viewing direction from the eye (or "camera") position, as shown in Figure, then this polygon is a back face if

$$\mathbf{V} \cdot \mathbf{N} > 0$$

Furthermore, if object descriptions have been converted to projection coordinates and our viewing direction is parallel to the viewing z,. axis, then $\mathbf{V} = (0, 0, V_z)$ and $\mathbf{V} \cdot \mathbf{N} = V_z C$

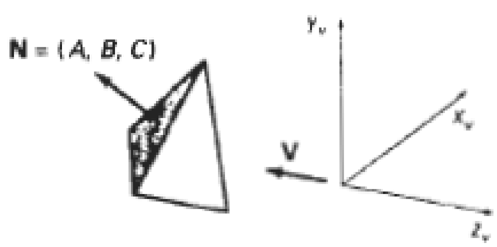
so that we only need to consider the sign of C, the ; component of the normal vector N.



Vector V in the viewing direction and a back-face normal vector N of a polyhedron

In a right-handed viewing system with viewing direction along the negative z axis (Figure), the polygon is a back face if C < 0. Also, we cannot see any face whose normal has z component C - 0, since our viewing direction is grazing that polygon. Thus, in general, we can label any polygon as a back face if its normal vector has a z-component value:

$$C \leq 0$$

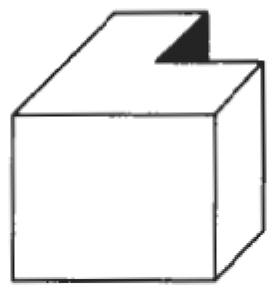


A polygon surface with plane parameter c<0 in a right hand viewing coordinate system is identified as a black face

Similar methods can be used in packages that employ a left-handed viewing system. In these packages, plane parameters A, B, C: and D can be calculated from polygon vertex coordinates specified in a clockwise direction (instead of the counterclockwise direction used in a right-handed system). Inequality 13-1 then remains a valid test for inside points.

Also, back faces have normal vectors that point away from the viewing position and are identified by $C \geq 0$ when the
Viewing direction is along the positive z_v axis.

By examining parameter C for the different planes defining an object, we can immediately identify all the back faces. For a single convex polyhedron, such as the pyramid in Fig. 3, this test identifies all the hidden surfaces on the object, since each surface is either completely visible or completely hidden. Also, if a scene contains only no overlapping convex polyhedral, then again all hidden surfaces are identified with the back-face method.



View of a concave polyhedron with one face partially hidden by other faces.

For other objects, such as the concave polyhedron in Figure, more tests need to be carried out to determine whether there are additional faces that are to ally or partly obscured by other faces. And a general scene can be expected to contain overlapping objects along the line of sight. We then need to determine where the obscured objects are partially or comp1etel.y hidden by other objects. In general, back-face removal can be expected to eliminate about half of the polygon surfaces in a scene from further visibility tests.

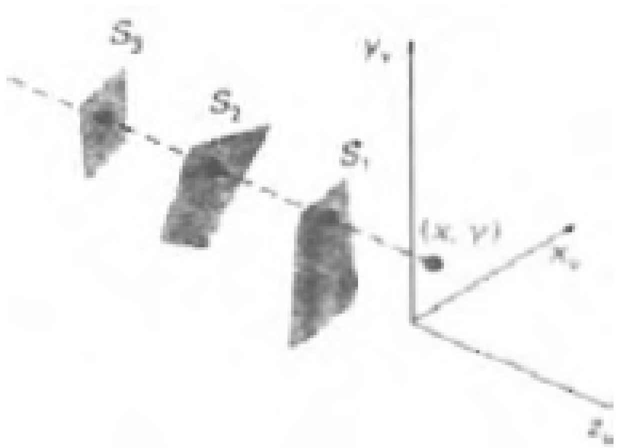
Depth Buffer Method

A commonly used image-space approach to detecting visible surfaces is the depth-buffer method, which compares surface depths at each pixel position on the projection plane. This procedure is also referred to as the z-buffer method, since object depth is usually measured from the view plane along the z axis of a viewing system. Each surface of a scene is processed separately, one point at a time across the surface. The method is usually applied to scenes containing only polygon surfaces, because depth values can be computed very quickly and the method is easy to implement. But the method can be applied to no planar surfaces.

With object descriptions converted to projection coordinates, each (x, y, z) position on a polygon surface corresponds to the orthographic projection point (x, y) on the view plane. Therefore, for each pixel position (x, y) on the view Plane, object depths can be compared by comparing z values. Figure4, shows three surfaces at varying distances along the orthographic projection line from position (x, y) in a view plane taken as the x_v,y_v, plane. Surface 5, is closest at this position, so its surface intensity value at (x, y) is saved.

We can implement the depth-buffer algorithm in normalized coordinates, so that *z* values range from 0 at the back clipping plane Z_{max} at the front clip ping plane. The value of Z_{max}, can be set either to 1 (for a unit cube) or to the largest value that can be stored on the system.

As implied by the name of this method, two buffer areas are required. A depth buffer is used to store depth values for each (x, y) position as surfaces are processed, and the refresh buffer stores the intensity values for each position. Initially, all positions in the depth buffer are set to 0 (minimum depth), and the refresh buffer is initialized to the background intensity. Each surface listed in the polygon tables is then processed, one scan line at a time, calculating the depth (z value) at each (x, y) pixel position. The calculated depth is compared to the value Previously stored in the depth buffer at that position. If the calculated depth is p a t e r than the value stored in the depth buffer, the new depth value is stored, and the surface intensity at that position is determined and in the same *xy* location in the refresh buffer.



At view-plane position (x, y), surface S, has the smallest depth from the view plane and so is visible at that position.

We summarize the steps of a depth-buffer algorithm as follows:

1. Initialize the depth buffer and refresh buffer so that for all buffer positions *(x*, y),

$$\text{depth}(x, y) = 0, \qquad \text{refresh}(x, y) = I_{\text{backgnd}}$$

2. For each position on each polygon surface, compare depth values to previously stored values in the depth buffer to determine visibility.

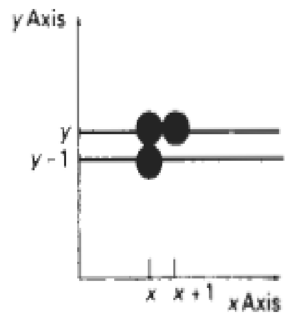
- Calculate the depth ***t*** for each *(x*, y) position on the polygon.
- If $z > \text{depth}(x, y)$, then set

$$\text{depth}(x, y) = z, \qquad \text{refresh}(x, y) = I_{\text{surf}}(x,y)$$

Where $l_{background}$ is the value for the background intensity, and $l_{surf(x,y)}$)is the projected intensity value for the surface at pixel position (x, y). After all surfaces have been processed, the depth buffer contains depth values for the visible surfaces and the buffer contains the corresponding intensity values for those surfaces.

Depth values for a surface position *(x*, y) are calculated from the plane equation for each surface:

$$z = \frac{-Ax - By - D}{C}$$



From position (x, y) on a scan line, the next position across the line has coordinates *(X* + 1, y), and the position immediately below on the next line has coordinates *(x,* y - 1).

For any scan line adjacent horizontal positions across the line differ by 1, and a vertical y value on an adjacent scan line differs by 1. If the depth of position *(x,* y) has been determined to be z, then the depth *z'* of the next position (x + 1, y) along the scan line is obtained

$$z' = \frac{-A(x+1) - By - D}{C}$$

Or

$$z' = z - \frac{A}{C}$$

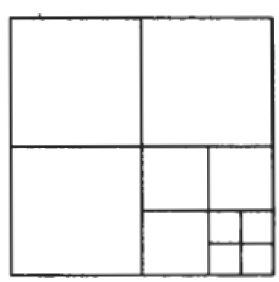
The ratio -*A/C* is constant for each surface, so succeeding depth values across a scan line are obtained from preceding values with a single addition.

Area Sub division Method

This technique for hidden-surface removal is essentially an image-space method, but object-space operations can be used to accomplish depth ordering of surfaces. The area-subdivision method takes advantage of area coherence in a scene by locating
those view areas that represent part of a single surface. We apply this method by successively dividing the total viewing area into smaller and smaller rectangles until each small area is the projection of part of n single visible surface or no surface at all.

To implement this method, we need to establish tests that can quickly identify the area as part of a single surface or tell us that the area is too complex to analyze easily. Starting with the total view, we apply the tests to determine whether we should subdivide the total area into smaller rectangles. If the tests indicate that the view is sufficiently complex, we subdivide it. Next. we apply the tests to each of the smaller areas, subdividing these if the tests indicate that visibility of a single surface is still uncertain. We continue this process until the subdivisions are easily analyzed as belonging to a single surface or until they are reduced to the size of a single pixel. An easy way to do this is to successively divide the area into four equal parts at each step, as shown in Fig. 13-20. This approach is similar to that used in constructing a quad tree. A viewing area with a resolution of 1024 by 1024 could be subdivided ten times in this way before a sub area is reduced to a pint.

Tests to determine the visibility of a single surface within a specified area are made by comparing surfaces to the boundary of the area. There are four possible relationships that a surface can have with a specified area boundary. We can describe these relative surface characteristics in the following way



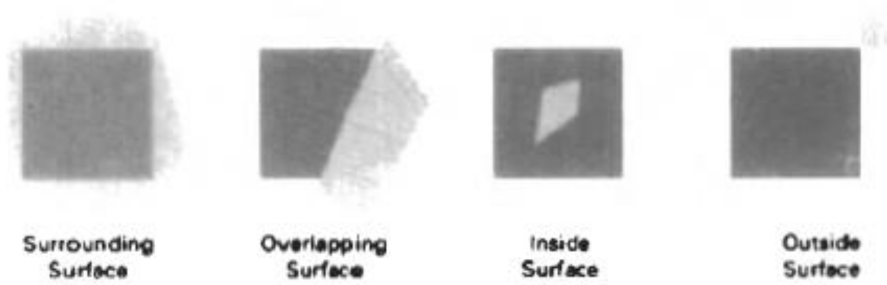
Dividing a square area into equal-sized quadrants at each

Step.

Surrounding surface-One that completely encloses the area.
Overlapping surface-One that is partly inside and partly outside the area.
Inside surface-One that is completely inside the area.
Outside surface-One that is completely outside the area.

The tests for determining surface visibility within an area can be stated in terms of these four classifications. No further subdivisions of a specified area are needed if one of the following conditions is true:

1. All surfaces are outside surfaces with respect to the area.
2. Only one inside, overlapping, or surrounding surface is in the area.
3. A surrounding surface obscures all other surfaces within the area boundaries.



POSSIBLE QUESTIONS
UNIT – V

Part-A

Online Examinations (One marks)

1. Duration of phosphorescence exhibited by the phosphor is called ______

   a)Pixel b) **Persistence** c) Electromagnetic d)CRT

2. Picture definition is stored in a memory area is called________

   a)Memory b)Picture definition c)**Frame Buffer** d)Fluorescent

3. The _____________in CRT is needed to force the electron beam to converge into a small spot as it strikes the phosphor.

   a) **Focusing system**b) Control grid c) Accessing point d) Pixel Position

4. An alphanumeric keyboard on a graphic system is used primarily as a device for entering

   a) **Text string** b) Numeric c) Alphanumeric d) String

5. ______ keys are common features on general purpose in keyboards

   a) Mouse b) Touch screen c) **Cursor-control key** d) Optical touch

6. The ________ key is used to co-ordinate position by positioning the screen cursor.

   a) **Cursor control key** b) Mouse c) Optical touch d) Function key

7. Another method for detecting mouse motion is _______

   a)Primary key b)Control grid c)Input **d)optical sensor**

8. Pixel positions along a line path are then determine by sampling at unit

   a)**x intervals** b)y intervals c)x and y d)x-y

9. An implementation of bresenham line drawing for slope in the range _____________ given in the following procedure

   a)m>1 b)**0<m<1** c)m<1 d)0>m>1

10. The call to______________ loads the intensity value 1 into the frame buffer at the specified (x,y)

    a) get pixel b)lines c)**set pixel** d)buffering

Part B

Essay Type (8 Marks)

1. What do you mean by back face removal?
2. Distinguish Graphics with Animation?
3. What are the techniques for 3D Transformation?
4. What do you mean by back face removal?
5. What do you mean by boundary points?
6. Discuss in detail the Depth-buffer method with its steps.
7. Write the method for translating, Rotation an object in 3Dgeometric Transformations.
8. Distinguish Parallel projection with that of Perspective projection.
9. Explain the concept of parallel projections With an Example.
10. Explain in Detail about Back Face Detection Method.

KARPAGAM ACADEMY OF HIGHER EDUCATION

COIMBATORE - 21

DEPARTMENT OF COMPUTER SCIENCE,CA & IT

CLASS : III B.Sc COMPUTER SCIENCE

BATCH : 2015-2018

Part -A Online Examinations **(1 mark questions)**

SUBJECT: COMPUTER GRAPHICS **SUBJECT CODE: 15CSU503**

Unit-V

S.no	Questions	option1	option2	option3	option4	Answer
1	The perception of color arises from __________ entering our visual system.	image	view volume	light	color	light
2	Light is a ________ energy.	electrical	mechanical	kinetic	electromagnetic	electromagnetic
3	The wavelength of the visual light ranges form _______nanometer.	100 to 500	400 to 700	30 to 90	600 to 1400	400 to 700
4	Which of these is the characteristics of light	brightness	hue	saturation	All the Above	All the Above
5	_________ coresponds to the physical property called luminance.	brightness	hue	saturation	purity	brightness
6	___________ property distinguishes a white light from a red or green light.	brightness	hue	saturation	purity	hue
7	_________ coresponds to the physical property called excitation purity.	brightness	hue	saturation	luminance	saturation
8	_________ describes the degree of vividness.	brightness	hue	saturation	luminance	saturation

9	Hue corresponds to another physical property called the ________	excitation purity	dominant wavelength	brightness	luminance	dominant wavelength
10	The receptor cells in the retina of the eye, that are sensitive to color is _________	rods	cones	cubes	lines	cones
11	The receptor cells in the retina of the eye, that are sensitive to color is _________	rods	cones	cubes	lines	rods
12	The international Commission on Illumination defined the _________ colour model.	RGB	CMY	XYZ	YIQ	XYZ
13	XYZ colors were the result of an ________________ transformation applied to three real primaries	Wavelet	Gamut	3 dimentional	Affine	Gamut
14	__________ of human eye coresponds to the eye's response to light of constant luminance.	dominant wavelength	luminous efficiency	excitation purity	None of the above	luminous efficiency
15	The curved triangular figure of CIE Chromaticity Diagram encompases _________	Whole Light spectrum	only the white light	all perceivable colors and	all perceivable colors by	all perceivable colors by ignoring
16	_______________ refers to NTSC	National Telecommunic	National Telecasting	National Television	National Tel-system	National Television System committee
17	______________Model , main focus is on the direct impact of the light coming from the light source	Global Illumination	Local Illumination	Specular Illumination	diffuse Illumination	Local Illumination
18	______________ Model, attempts to include secondary effects as light going through transparent / translucent	Global Illumination	Specular Illumination	Local Illumination	diffuse Illumination	Global Illumination
19	In _____________ reflection, light energy from the light source gets reflectd / bounced off equally in all the	Global reflection	Local reflection	Specular reflection	diffuse reflection	diffuse reflection
20	________________ reflection, attempts to capture the characteristics of a shiny or mirror-like surface	diffuse reflection	Local reflection	Specular reflection	Global reflection	Specular reflection
21	In ___________ shading, Instead of color values, normal vector is found Interpolatively	Phong	constant	Gouraud	sutherland	Phong
22	Regular or Irregular surface feature details are colectively referred to as _________	watermark texture	glomming texture	surface texture	scaleable texture	surface texture

23	When an AND operation is used, a texture area with _________ shades will appear unaltered if the original	black	white	red	magenta	magenta
24	When an AND operation is used, a texture area with _________ shades will appear unaltered if the original	black	white	red	magenta	red
25	_________ texture is an effective tool when target surface are relatively flat and facing the reference plane	projected	solid	mapping	interpolative	projected
26	In _______________ texture, we can wrap around the surface of an object, stretch or shrink it so as to follow the	projected	solid	texture mapping	interpolative	texture mapping
27	__________ is a 3D representation of the internal structure of some nonhomogeneous material.	projected	solid	texture mapping	interpolative	solid
28	A Global illumination model that accounts for the transport of light energy beyond the direct contribution	Light Tracing	Ray Tracing	basic Tracing	none of the above	Ray Tracing
29	Which among the following is NOT, the three components of the surface shading used in several	local	reflected	transmitted	refracted	refracted
30	A Vector is defined by its	direction and starting point	direction and end point	only direction	direction and magnitude	direction and magnitude
31	A Ray is determined by its	direction and starting point	direction and end point	only direction	direction and magnitude	direction and starting point
32	___________ contribution refers to the direct contribution from the light source	refracted	reflected	transmitted	local	local
33	___________ contribution refers to the reflection of light energy coming from aother object surface	reflected	refracted	transmitted	local	reflected
34	___________ contribution refers to the transmission of light energy coming from behind the surface	reflected	refracted	transmitted	local	transmitted
35	a Family of vector are calculated using the formulae	|s + td - c |	s + td	| s-c |	|s - td - c |	s + td
36	An Opaque object ___________ light	transmit	pass through	does not transmit	does not pass through	does not transmit

37	The value of the reflection coefficents of the objects along the path of reflection - Kr1, Kr2, Kr3	0.01	0.1	0.2	2	0.1
38	Dull object ________ specular reflection	transmit	does not transmit	produce	does not produce	does not produce
39	Purpose of Bounding Volume Extension technique is to identify object, especially _________ object	simple	complex	curved	flat	complex
40	___________ technique is based on the observation that only objects that are in the path of a ray may intersect by	bounding volume	adaptive depth control	hierarchy of bounding	spatial coherence	spatial coherence
41	The vector equation for the sphere is ___________	| s-c |	|p -c|	|d-c|	|s-t|	| s-c |
42	In ____ sampling, a separate primary ray is sent and traced through the center of each subpixel	spatial	stochastic	super	adaptive flter	super
43	___________ send one ray through the center of a pixel and four additional rays through its corners	Adaptive supersampling	stochastic	super	adaptive flter	Adaptive supersampling
44	In _____________ ray deviate from using the fixed pixel grid by scattering th erays evenly across the pixel area.	Adaptive supersampling	stochastic supersampling	super	adaptive flter	stochastic supersampling
45	__________ are then distributed to these zones via random selection	soft shadow	burry reflection	transulency	motion blur	soft shadow
46	The distribution of the angle is subject to the same ___________ reflectance function the governs hightlights	circle-shaped	rounded-shaped	star shaped	bell-shaped	bell-shaped

Reg. No................................

[10CSU504]

KARPAGAM UNIVERSITY

(Under Section 3 of UGC Act 1956)
COIMBATORE – 641 021
(For the candidates admitted from 2010 onwards)

B.Sc. DEGREE EXAMINATION, NOVEMBER 2012

Fifth Semester

COMPUTER SCIENCE

COMPUTER GRAPHICS

Time: 3 hours Maximum : 60 marks

PART – A (20X ½ = 10 Marks)
Answer ALL the Questions

1. Animations are often used in
a) CAD b) Image processing c)Computer Art d) laser scan

2. Mathematical software is used to create ______
a) 3D image Pairs b)art c) Animations d) an image

3. Shadow-mask methods are commonly used in
a) random scan b) raster scan c) display processor d) refresh CRT.

4. The term liquid crystals refers to
a) atoms b)electrons c) molecules d) Phosphor.

5. "Slices" of objects can be displayed using
a) refresh CRT b)3D Viewing Devices c) Random Scan d) Raster Scan

6. The input device used for entry of screen coordinates is
a) Keyboard b)Mouse c) TrackBall d)Touch Panels

7 Trackball is a ______
a) 2D pointing device b) 3D pointing device c) input device d) output device

8. Some joysticks are mounted on a ______
a) Mouse b) Digitizers c) Image scanners d) keyboard

9. Optical scanning mechanism is used in ______.
a) Mouse b) Trackball c) Image scanners d) voice systems

10. Frequency pattern is established in ______
a) Raster Scan b) Trackball c) light pens d) voice systems.

11. These devices display a straight line segment by plotting discrete points
a) Raster graphics devices b) Digital devices c) input devices
d) CRT Monitor

12. In the Cartesian slope intercept equation for a straight line m represents ______.
a) endpoints b)slope c) intercept d)intervals

13. It is a faster method for calculating pixel positions
a) Line Function b)Line Drawing c) DDA algorithm
d)Bresenham's Line algorithm

14. To apply Bresenham's algorithm over the partitions we need the initial value for ______.
a)x coordinate b)y coordinate c)both d) z coordinate

15. In a mid point circle algorithm, any point (x,y) on the boundary of the circle with radius r satisfies the equation
a)$f_{circle}(x,y)=1$ b) $f_{circle}(x,y)=2$ c) $f_{circle}(x,y)=3$ d) $f_{circle}(x,y)=0$

16. Forming products of transformation matrices is referred to as a
a) Translations b) Rotations c) Composition d) Location

17. In this process lines through the "inside-outside" tests by checking the line end points.
a) Point clipping b) Line Clipping c) Text Clipping d) Area Clipping

18. In this clipping line clipper is displayed as a series of unconnected line segments?
a) Point clipping b) Line Clipping c)Polygon Clipping d) Area Clipping

19. This technique is used in engineering and architectural drawings
a) Surface Rendering b) Parallel projection c) Perspective projection
d) Transalations.

20. This method takes advantage of area coherence in a scene by locating those views areas represent part of a single surface.
a) Parallel b) Perspective c) area-subdivision d) Axonometric

PART B (5 X 4= 20 Marks)
Answer ALL the Questions

21. a. What is the use of Random scan display device.

Or

b. Explain various activities of Flat panel Displays.

22. a. Listout the functions of Input Device mouse.

Or

b. What is the purpose of the image scanners?

23. a. Explain the function of the Midpoint circle algorithm?

Or

b. Clearly Explain about General Fixed point scaling.

24. a. How to view co-ordinate reference Name?

Or

b. Explain the function of 'Polygon clipping?

25. a. Discuss the term 'Parallel projection' in Detail?

Or

b. Explain the term 'Book Face Detective'

PART C (3 x 10 = 30 Marks)
Answer any THREE Questions

26. What are the activities of Graphics monitors and work stations?

27. Briefly explain the various usage of Digitizers.

28. List the various steps in DDA Algorithm.

29. Explain the term 'Viewing pipeline'?

30. Explain various procedures in three dimensional Geometric transformations?

Reg. No................................

[11CSU603]

KARPAGAM UNIVERSITY

(Under Section 3 of UGC Act 1956)

COIMBATORE – 641 021

(For the candidates admitted from 2011 onwards)

B.Sc. DEGREE EXAMINATION, APRIL 2014

Sixth Semester

COMPUTER SCIENCE

COMPUTER GRAPHICS

Time: 3 hours | Maximum : 100 marks

PART – A (15 x 2 = 30 Marks)
Answer ALL the Questions

1. List the various applications of Computer Graphics?
2. Define persistence.
3. Mention any two drawbacks of DVST.
4. Which Input device is directly used on monitor and how it has been done
5. Differentiate between Input and Output device with one example
6. Define plotter.
7. Mention the steps used in DDA line drawing algorithm.
8. Give the matrix representation of 2D transformation
9. Write the properties of Circles
10. How the window to view-port transformation is performed
11. Give a note on Point clipping
12. How the text are clipped in text clipping process
13. Write the transformation equations of 3D
14. List some Visible Surface Detection Algorithms
15. Give a brief note on depth buffer method.

PART B (5 X 14= 70 Marks)
Answer ALL the Questions

16. (a) Discuss about Color CRT Monitors.

Or

(b) Write the working principles of Refresh CRT and a detail Description of Each component in CRT.

17. (a) Give a short note of the following Input Devices
i. Digitizers ii. Data glove iii. Track ball and Space ball

Or

(b) Give a short note of the following Output Devices
i. Printer ii. Plotter

18. (a) Explain in detail of Bresenham's Line Drawing algorithm.

Or

(b) Describe about 2D Transformation and composite Transformation.

19. (a) How the polygon clipping process takes place in Sutherland-Hodgman polygon clipping algorithm.

Or

(b) How the line clipping process happens in Cohen Sutherland Line Clipping Algorithm.

20. (a) Discuss about various 3D Geometric Transformation display methods.

Or

(b) Explain in Detail about Perspective Projection with neat diagram.

Reg. No...............................

[12CSU603]

KARPAGAM UNIVERSITY

(Under Section 3 of UGC Act 1956)
COIMBATORE – 641 021
(For the candidates admitted from 2012 onwards)

B.Sc., DEGREE EXAMINATION, APRIL 2015

Sixth Semester

COMPUTER SCIENCE

COMPUTER GRAPHICS

Time: 3 hours | Maximum : 100 marks

PART – A (15 x 2 = 30 Marks)
Answer ALL the Questions

1. Define the computer graphics
2. What is called flat panel display?
3. What is the use of 3 Dimensional viewing devices?
4. What is track ball?
5. What is a joystick?
6. Define Printer?
7. What is print clipping?
8. What is translation?
9. Write about Scaling?
10. What is called point?
11. Define Polygon Clipping
12. What is the purpose of viewing pipeline?
13. Define parallel projection
14. What is back face detection?
15. Write about Depth Butter method?

PART B (5 X 14= 70 Marks)
Answer ALL the Questions

16. a. Give a note on color CRT monitors.

Or

b. Briefly explain about Three Dimensional viewing devices

17. a. Explain about Input Device.

Or

b. Explain the working principle of different types of printes.

18. a. Explain about the Viewing Pipeline concept.

Or

b. Write the algorithm of Sutherland Hodgeman Polygon clipping and explain it with an example.

19. a. Discuss about 3D Geometric transformations

Or

b. Explain about Depth Buffer method in details.

20. **Compulsory** : -

Compulsory- Give a note on composite transformation in details

----