B.E.

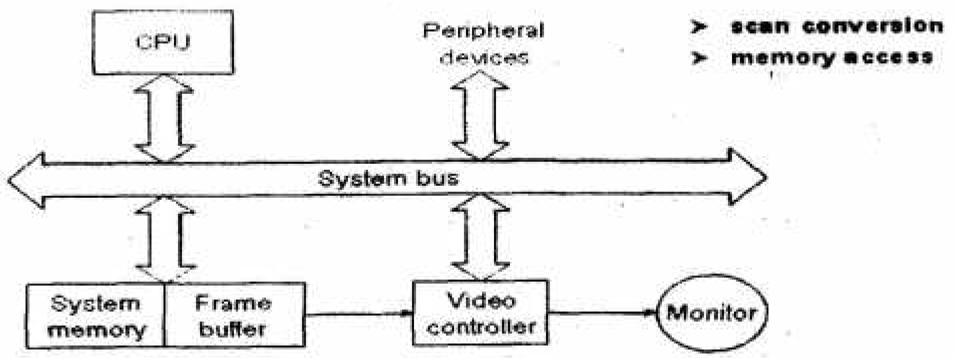
Fifth Semester Examination, Dec-2008

COMPUTER GRAPHICS

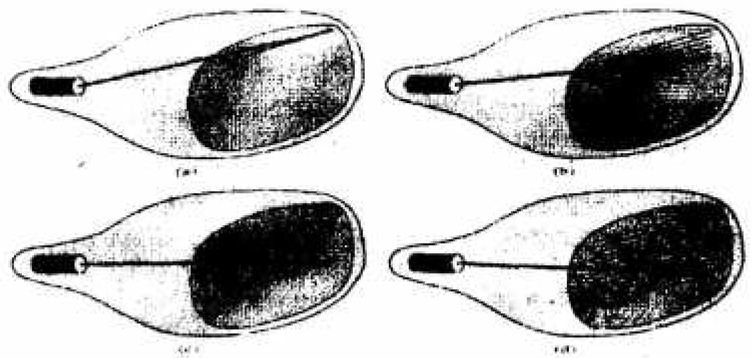
Note : Attempt any five questions.

Q. 1. (a) Draw the block diagram and explain the working of a raster display system? Differentiate between horizontal retrace and vertical retace.

Ans. Block Diagram :

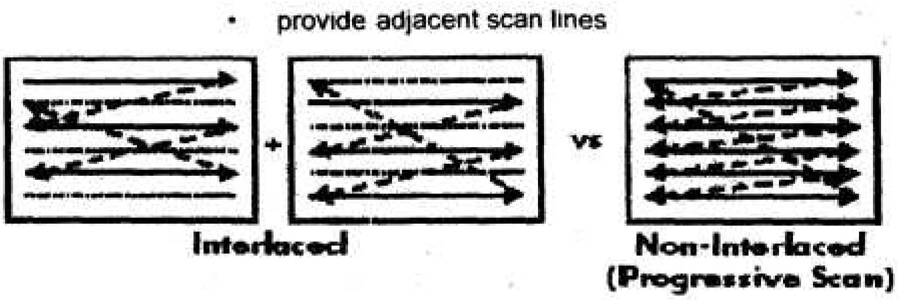


- Point plotting device: pixel or pel (Picture Element).
- Picture info. for all the screen points is stored in separate Memory called Frame (Refresh Buffer).
- One row at a time (scan line) from top to bottom.



■ Frame buffer

* **Frame buffer :**
  - Depth of the buffer area, Number of bit planes
  - Bitmap : one bit per pixel
  - Pixmap : multiple bits per pixel.
* **Refresh rate :**
  - Above about 24 frame per second
  - Unit of refresh rates Hz
* Ex) 60 frames per second (60Hz)
* **Retrace**
  - Horizontal retrace Nertical retrace.
  - Interlaced refresh procedure.
  - Used with slower refreshing rates
* Avoid flicker
* Provide adjacent scan lines.



Q. 1. (b) Write the steps required to plot a line whose slope is between 0° and 45° using Bresenham's method.

Ans. Line drawing algorithm : Steps are as follows :

BRESENHAM'S LINE DRAWING ALGORITHM

(for lml < 1.0)

1. Input the two line end-points, storing the left end-point in (x0, y0).
2. Plot the point (x0, y0).
3. Calculate the constants Δx, Δy, $2\Delta y$ and $(2\Delta y - 2\Delta x)$ and get the first value for the decision parameter as:

1. At each xk along the line, starting at k = 0, perform the following test. If pk < 0, the next point to plot is (xk+1, yk) and :

$$p_{k+1} = p_k + 2\Delta y$$

Otherwise, the next point to plot is $(xk + 1, yk + 1)$ and :

$$p_{k+1} = p_k + 2\Delta y - 2\Delta x.$$

5. Repeat step $4(\Delta x - 1)$ times.

Q. 2. Briefly explain the following display techniques :

(i) CRT

(ii) LCD

(iii) Plasma.

Ans. CRT : The cathode ray tube (CRT) is a vacuum tube containing an electron gun (a source of electrons) and a fluorescent screen, with internal or external means to accelerate and deflect the electron beam, used to form images in the form of light emitted from the fluorescent screen. The image may represent electrical waveforms (oscilloscope), pictures (television, computer monitor), radar targets and others. The single electron beam can be processed in such a way as to display moving pictures in natural colors. The CRT uses an evacuated glass envelope which is large, deep, heavy, and relatively fragile. Display technologies without these disadvantages, such as flat plasma screens, liquid crystal displays, DLP, OLEO displays have replaced CRTs in many applications and are becoming increasingly common as costs decline.

The cathode rays are now known to be a beam of electrons emitted from a heated cathode inside a vacuum tube and accelerated by a potential difference between this cathode and an anode. The screen is covered with a phosphorescent coating (often transition metals or rare earth elements), which emits visible light when excited by high-energy electrons. The beam is deflected either by a magnetic or an electric field to move the bright dot to the required position on the screen.

In television sets and computer monitors the entire front area of the tube is scanned systematically in a fixed pattern called a raster. An image is produced by modulating the intensity of the electron beam with a received video signal (or another signal derived from it). In all CRT TV receivers except some very early models, the beam is deflected by magnetic deflection, a varying magnetic heid generated by coils (the magnetic yoke), driven by electronic circuits, around the neck of the tube.

The source of the electron beam is the electron gun, which produces a stream of electrons through thermionic emission, and focuses it into a thin beam. The gun is located in the narrow, cylindrical neck at the extreme rear of a CRT and has electrical connecting pins, usually arranged in a circular configuration, extending from its end. These pins provide external connections to the cathode, to various grid elements in the gun used to focus and modulate the beam, and, in electrostatic deflection CRTs, to the deflection plates. Since the CRT is a hot-cathode device, these pins also provide connections to one or more filament heaters within the electron gun. When a CRT is operating, the heaters can often be seen glowing orange through the glass walls of the CRT neck. The need for these heaters to 'warm up' causes a delay between the time that a CRT is first turned on, and the time that a display becomes visible. In older tubes, this could take fifteen seconds or more; modern CRT displays have fast starting circuits which produce an image within about two seconds, using either briefly increased heater current or elevated cathode voltage. Once the CRT has warmed up, the heaters stay on continuously. The electrodes are often covered with a black layer, a patented process used by all major CRT manufacturers to improve electron density.

The electron gun accelerates not only electrons but also ions present in the imperfect vacuum (some of which result from outgassing of the internal tube components). The ions, being much heavier than electrons, are deflected much less by the magnetic or electrostatic fields used to position the electron beam. Ions striking the screen damage it; to prevent this the electron gun can be positioned slightly off the axis of the tube so that the ions strike the side of the CRT instead of the screen. Permanent magnets (the ion trap) deflect the lighter electrons so that they strike the screen. Some very old TV sets without an ion trap show browning of the center of the screen, known as ion burn. The aluminum coating used in later CRTs reduced the need for an ion trap.

When electrons strike the poorly-conductive phosphor layer on the glass CRT, it becomes electrically charged, and tends to repel electrons, reducing brightness (this effect is known as "sticking"). To prevent this the interior side of the phosphor layer can be covered with a layer of aluminum connected to the conductive layer inside the tube, which disposes of this charge. It has the additional advantages of increasing brightness by reflecting towards the viewer light emitted towards the back of the tube, and protecting the phosphor from ion bombardment.

(ii) LCD :

A liquid crystal display (LCD) is an electro-optical amplitude modulator realized as a thin, flat display device made up of any number of color or monochrome pixels arrayed in front of a light source (backlight) or reflector. It is often utilized in battery-powered electronic devices because it uses very small amounts of electric power. A comprehensive classification of the various types and electro-optical modes of LCDs is provided in the article LCD classification. Each pixel of an LCD typically consists of a layer of molecules aligned between two transparent electrodes, and two polarizing filters, the axes of transmission of which are (in most of the cases) perpendicular to each other. With no actual liquid crystal between the polarizing filters, light passing through the first filter would be blocked by the second (crossed) polarizer.

The surface of the electrodes that are in contact with the liquid crystal material are treated so as to align the liquid crystal molecules in a particular direction. This treatment typically consists of a thin polymer layer that is unidirectionally rubbed using, for example, a cloth. The direction of the liquid crystal alignment is then defined by the direction of rubbing. Electrodes are made of a transparent conductor called Indium Tin Oxide (ITO).

Before applying an electric field, the orientation of the liquid crystal molecules is determined by the alignment at the surfaces. In a twisted nematic device (still the most common liquid crystal device), the surface alignment directions at the two electrodes are perpendicular to each other, and so the molecules arrange themselves in a helical structure, or twist. Because the liquid crystal material is birefringent, light passing through one polarizing filter is rotated by the liquid crystal helix as it passes through the liquid crystal layer, allowing it to pass through the second polarized filter. Half of the incident light is absorbed by the first polarizing filter, but otherwise the entire assembly is reasonably transparent. When a voltage is applied across the electrodes, a torque acts to align the liquid crystal molecules parallel to the electric field, distorting the helical structure (this is resisted by elastic forces since the molecules are constrained at the surfaces). This reduces the rotation of the polarization of the incident light, and the device appears grey. If the applied voltage is large enough, the liquid crystal molecules in the center of the layer are almost completely untwisted and the polarization of the incident light is not rotated as it passes through the liquid crystal layer. This light will then bA mainly polarized perpendicular to the second filter, and thus be blocked and the pixel will appear black. By controlling the voltage applied across the liquid crystal layer in each pixel, light can be allowed to pass through in varying amounts thus constituting different levels of gray.

(iii) Plasma :

Plasma is a partially ionized gas, in which a certain proportion of electrons are free rather than being bound to an atom or molecule. The ability of the positive and negative charges to move somewhat independently makes the plasma electrically conductive so that it responds strongly to electromagnetic fields. Plasma therefore has properties quite unlike those of solids, liquids or gases and is considered to be a distinct state of matter. Plasma typically takes the form of neutral gas-like clouds, as seen, for example, in the case of stars. Like gas, plasma does not have a definite shape or a definite volume unless enclosed in a container, but unlike gas, in the influence of a magnetic field, it may form structures such as filaments, beams and double layers. Plasmas are by far the most common phase of matter in the universe, both by mass and by volume. All the stars are made of plasma, and even the space between the stars is filled with a plasma, albeit a very sparse. In our solar system, the planet Jupiter accounts for most of the non-plasma, only about 0.1 % of the mass and 10-15% of the volume within the orbit of Pluto. Very small grains within a gaseous plasma will also pick up a net negative charge, so that they in turn may act like a very heavy negative ion component of the plasma.

Q. 3. (a) Derive the transformation that rotates an object point $\theta°$ about the origin. Write the matrix representation for this.

Ans. Transformation of Rotation of Objects :

* The overall transformation is a combination of three elementary ones: scaling, rotation and translation.

* Transformations are useful in a number of situations :

1. Composing a "scene" out of a number of objects.

2. Creating a complex object from a single "motif".

3. A designer may to view an object from different vantage points and make a picture from each one.

4. In a computer animation, several objects must move to one another from frame to frame.

*** Object Transformations Versus Coordinate Transformations :**

- There are two ways to view a transformation: as an object transformation or as a coordinate transformation.

- An object transformation alters the coordinates of each point on the object to some rule, leaving the underlying coordinate system fixed.

A coordinate transformation defines a new coordinate system in terms of the old one and then represents all of the object's points in this new system.

1. Transforming Points and Objects :

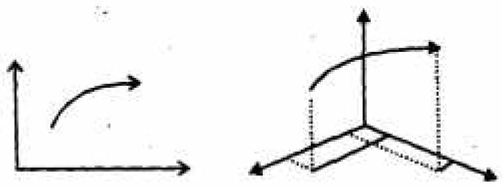
A transformation alters each point P in space (20 or 3D) into a new point Q by means of a specific formula or algorithm.

$$P = (P_{wx}, P_{wy}, 1)$$

$$Q = (Q_{wx}, Q_{wy}, 1)$$

$$(Q_{wx}, Q_{wy}, 1) = T(P_{wx}, P_{wy}, 1)$$

$$Q = T(P)$$



2. The Affine Transformations :

- Affine transformations are the most common transformations used in computer graphics.
- Affine transformations have a simple form: The coordinates of Q are linear combinations of those of P. That is,

$$(Qx, Qy, 1) = (m11Px + m12Py + m13, m11Px + m12Py + m13, 1)$$

$$\begin{pmatrix} Q \\ Q \\ 1 \end{pmatrix} = \begin{bmatrix} m & m & m \\ m & m & m \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} P \\ P \\ 1 \end{pmatrix}$$

For an affine transformation, the third row of the matrix is always (0, 0, 1)

Geometric Effects of Elementary 20 Affine Transformations

- Affine transformations produce combinations of four elementary transformations: translation, scaling, rotation, and shear.

(a) Translation :

$$\begin{pmatrix} Q \\ Q \\ 1 \end{pmatrix} = \begin{bmatrix} 1 & 0 & dx \\ 0 & 1 & dy \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} P \\ P \\ 1 \end{pmatrix}$$

(b) Scaling :

$$\begin{pmatrix} Q \\ Q \\ 1 \end{pmatrix} = \begin{bmatrix} S & 0 & 0 \\ 0 & S & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} P \\ P \\ 1 \end{pmatrix}$$

(c) Rotation :

$$\begin{pmatrix} Q \\ Q \\ 1 \end{pmatrix} = \begin{bmatrix} \cos(a) & -\sin(a) & 0 \\ \sin(a) & \cos(a) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} P \\ P \\ 1 \end{pmatrix}$$

(d) Shearing :

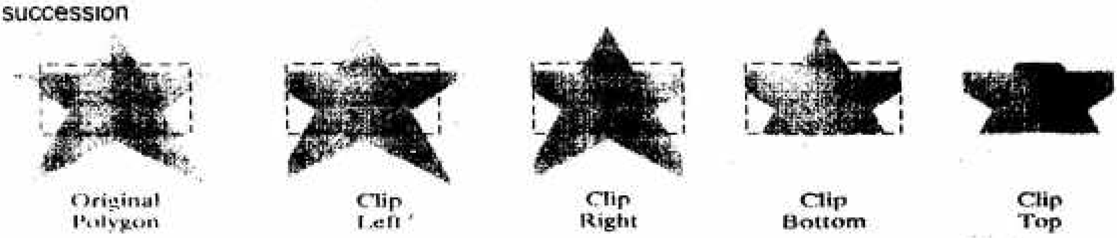
$$\begin{pmatrix} Q \\ Q \\ 1 \end{pmatrix} = \begin{bmatrix} 1 & h & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} P \\ P \\ 1 \end{pmatrix}, \begin{pmatrix} Q \\ Q \\ 1 \end{pmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ g & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} P \\ P \\ 1 \end{pmatrix}$$

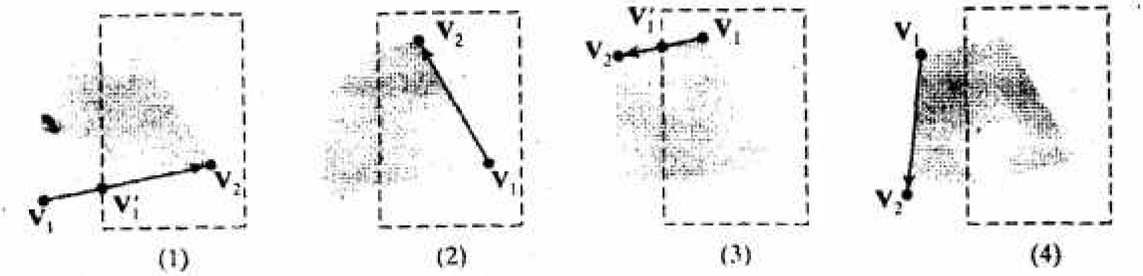
Q. 3. (b) Write and explain briefly the Sutherland Hodgeman polygon clipping algorithm.

Ans. Sutherland Hodgeman Polygon clipping algorithm : Clip against 4 infinite clip edges in succession

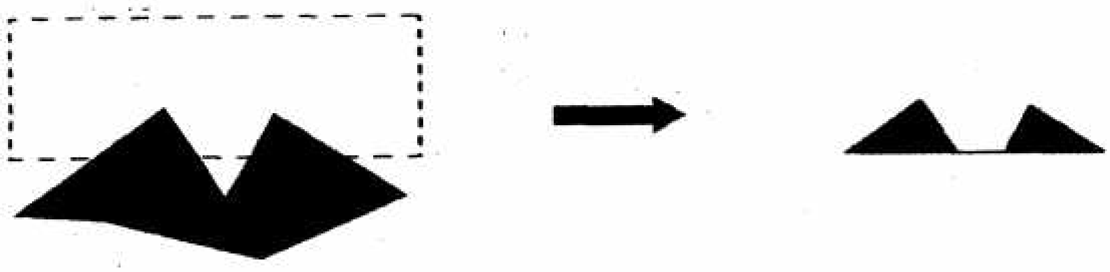
- Accept a series of vertices (polygon) and outputs another series of vertices
- Four possible outputs.

succession





- The algorithm correctly clips convex polygons, but may display extraneous lines for concave polygons.



Q. 4. (a) Show what happens when an object "behind" the centre of projection is projected by $M_{perspective}$ and then clipped why in general one can't project and then clip.

Ans. Given the specification of the view window, we can set up a view volume using the window boundaries. Only those objects within the view volume will" appear in the generated display on an output device; all others are clipped from the display. The size of the view volume depends on the size of the window, while the shape of the view volume depends on the type of projection to be used to generate the display. In any case, four sides of the volume are planes that pass through the edges of the window. For a parallel projection, these four sides of the view volume form an infinite parallelepiped. For a perspective projection, the view volume is- a with apex at the projection reference point.

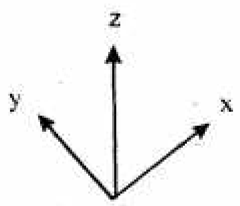
A finite view volume is obtained by limiting the extent of the volume in the 2,. direction. This is done by specifying positions for one or two additional boundary planes. These z-boundary planes are referred to as the front plane and back plane, or the near plane and the far plane, of the viewing volume. The front and back planes are parallel to the view plane at specified-positions and z"". Both planes must be on the same side of the projection reference point, and the back plane must be farther from the projecting point than the front plane. Including the front and back planes produces a view volume bounded by six planes, With an orthographic parallel projection, the six planes form a rectangular parallelepiped, while an oblique parallel projection produces an oblique parallelepiped view volume. With a perspective projection, the front and back clipping planes truncate the infinite pyramidal view volume to form a frustum.

Front and back clipping planes allow us to eliminate parts of the scene from the viewing operations based on depth. We can then pock out parts of a scene that we would like to view and exclude objects that are 111 front of or behind the part that we want to look at. Also, in a perspective projection, we can use the front clipping plane to take out large objects close to the view plane that can project into unrecognazable sections within the view window. Similarly, the back clip- ping plane can he used to cut out objects far from the projection reference point that can project to small blotson the output devict* reference system.

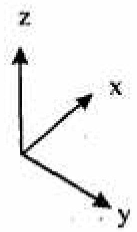
Q. 4. (b) What do you mean by Co-ordinate system? Explain normalized device co-ordinate system and logical co-ordinate system.

Ans. Co-ordinate systems :

A system for specifying points using coordinates measured in some specified way. The simplest coordinate system consists of coordinate axes oriented perpendicularly to each other, known as Cartesian coordinates. Depending on the type of problem under consideration, coordinate systems possessing special properties may allow particularly simple solution.



Right-handed Coordinate System



Left-handed Coordinate System

In three dimensions, so-called right-handed coordinate systems (left figure) are usually chosen by convention, although left-handed coordinate systems (right figure) are also encountered occasionally.

Q. 5. Why we require hidden surface removal algorithms? Write and explain the scan-line method for hidden surface removal? How the amount of computation can be reduced in this method?

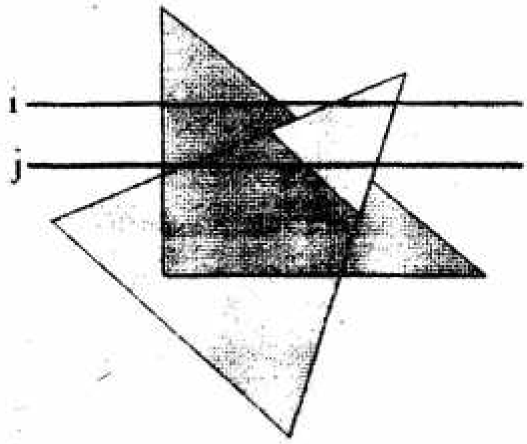
Ans. Hidden Surface removal algorithm :

- Visible surface determination
- Object space and image space approaches
- 2 polygons, different sizes and shapes
- Four possibilities
- A completely obscure B, display only A
- B obscures A , display only B
- A and B both are completely visible
- A and B partially obscure each other, calculate the visible parts of each polygon
- In 3d we must be concerned with whether or not objects are obscured by other objects.
- Most objects are opaque so should obscure things behind them.
- A. K. A visible surface detection methods or hidden surface elimination methods.

Scan- Line Algorithm :

- Similar in some respects to the z-buffer method but handles the image scan-line by scan-line
- Rasterize all polygon boundaries (edges).
- Scanning across each scan line we determine the colour of each pixel
- By default colour everything as background.
- If we encounter the edge of one polygon, start evaluating polygon colour* at each point and shade the scanline it accordingly.

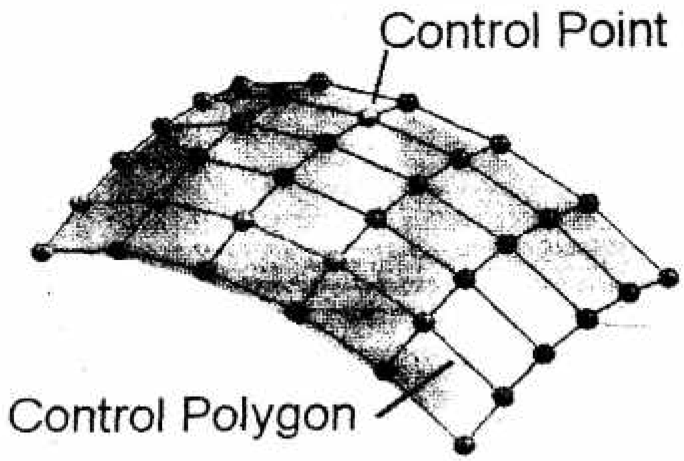
- For multiple edges do depth evaluation to see which polygon "wins".
- Due to coherency in data, this can be relatively efficient.



Q. 6. (a) Where has the term "Spline" originated from? Differentiate between uniform non-rational B-spline and non-uniform non-rational B-spline with suitable examples.

Ans. Spline :

Splines are popular curves in these subfields because of the simplicity of their construction, their ease and accuracy of evaluation, and their capacity to approximate complex shapes through curve fitting and interactive curve design. The term "spline" is used to refer to a wide class of functions that are used in applications requiring data interpolation and/or smoothing. The data may be either one-dimensional or multi-dimensional. Spline functions for interpolation are normally determined as the minimizers of suitable measures of roughness (for example integral squared curvature) subject to the interpolation constraints. Smoothing splines may be viewed as generalizations of interpolation splines where the functions are determined to minimize a weighted combination of the average squared approximation error over observed data and the roughness measure. For a number of meaningful definitions of the roughness measure, the spline functions are found to be finite dimensional in nature, which is the primary reason for their utility in computations and representation A NURBS curve is defined by its order, a set of weighted control points, and a knot vector. NURBS curves and surfaces are generalizations of both B-splines and Bezier curves and surfaces, the primary difference being the weighting of the control points which makes NURBS curves rational (non-rational B-splines are a special case of rational B-splines). Whereas NURBS curves evolve into only one parametric direction, usually called s or u, NURBS surfaces evolve into two parametric directions, called sand for U and v.



By evaluating a NURBS curve at various values of the parameter, the curve can be represented in Cartesian two- or three-dimensional space. Likewise, by evaluating a NURBS surface at various values of the two parameters, the surface can be represented in Cartesian space.

NURBS curves and surfaces are useful for a number of reasons :

- They are invariant under affine as well as perspective transformations: operations like rotations and translations can be applied to NURBS curves and surfaces by applying them to their control points.
- They offer one common mathematical form for both standard analytical shapes (e.g., cónics) and free-form shapes. They provide the flexibility to design a large variety of shapes.
- They reduce the memory consumption when storing shapes (compared to simpler methods).
- They can be evaluated reasonably quickly by numerically stable and accurate algorithms.

Q. 6. (b) Briefly explain any method for displaying bicubic surfaces.

Ans. Bicubic surface :

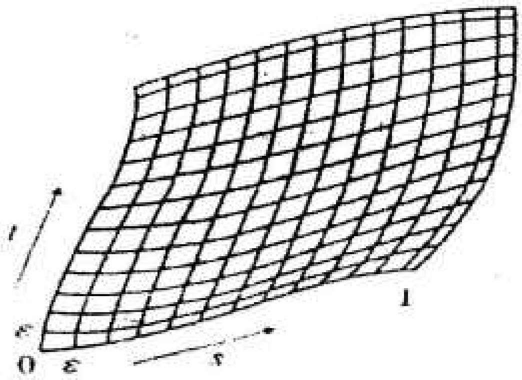
$$f(t) = at^3 + bt^2 + ct + d$$

$$f(t+\delta) = f(t) + \Delta f(t)$$

$$\Delta f(t) = f(t+\delta) - f(t)$$

$$= 3a\delta t^2 + \left(3a\delta^2 + 2b\delta\right)t + a\delta^3 + b\delta^2 + c\delta$$

/



Similarly,

$$\Delta^2 f(t) = 6a\delta^2 t + 6a\delta^3 + 2b\delta^2$$

$O\left(\left(\frac{1}{\varepsilon}\right)^2\right)$

$$\Delta^3 f(t) = 6a\delta^3$$

$$f(t+\delta) = f(t) + \Delta f(t)$$

$$t_i = \delta, i, v$$

Where

$$n = \frac{1}{\delta}$$

$$f_i = f(t_i)$$

$$\Delta f_i = \Delta f(t_i)$$

$$\Delta^2 f_i = \Delta^2 f(t_i)$$

$$\Delta^3 f_i = \Delta^3 f(t_i)$$

$$f(t_{i+1}) = f(t_i + \delta) = f(t_i) + \Delta f(t_i)$$

$$\left.\begin{aligned} f_{i+1} &= f_i + \Delta f_i \\ \Delta f_{i+1} &= \Delta f_i + \Delta^2 f_i \\ \Delta^2 f_{i+1} &= \Delta^2 f_i + \Delta^3 f_i \end{aligned}\right\} i \geq 0$$

$$\begin{bmatrix} f_0 \\ \Delta f_0 \\ \Delta^2 f_0 \\ \Delta^3 f_0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 1 \\ \delta^3 & \delta^2 & \delta & 0 \\ 6\delta^3 & 2\delta^2 & 0 & 0 \\ 6\delta^3 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix}$$

$$f_{i+1} = f_i + \Delta f_i$$

$$\Delta f_{i+1} = \Delta f_i + \Delta^2 f_i$$

$$\Delta^2 f_{i+1} = \Delta^2 f_i + \Delta^3 f_i$$

$$f(t) = at^3 + bt^3 + ct + d$$

$$\Delta f(t) = 3a\delta t^3 + \left(3a\delta^2 + 2b\delta\right)t + a\delta^3 + b\delta^2 + c\delta$$

$$\Delta^2 f(t) = 6a\delta^2 t + 6a\delta^3 + 2b\delta^2$$

$$\Delta^3 f(t) = 6a\delta^3$$

$\therefore$

$$f_0 = d$$

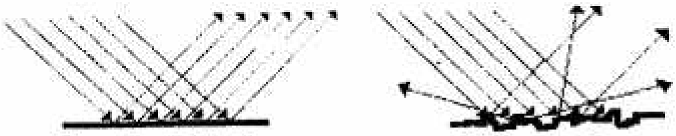
$$\Delta f_0 = a\delta^3 + b\delta^2 + c\delta$$

$$\Delta^2 f_0 = 6a\delta^3 + 2b\delta^2$$

$$\Delta^3 f_0 = 6a\delta^3.$$

Q. 7. Differentiate between diffuse and specular reflection? Also explain Phong illumination model.

Ans. Diffuse and specular reflection the behavior of light is often studied by observing its reflection off of plane (flat) mirrors. Mirrors are typically smooth surfaces, even at the microscopic levels. As such, they offer each individual ray of light the same surface orientation. But quite obviously, mirrors are not the only types of objects which light reflects off of. Most objects which reflect light are not smooth at the microscopic level. Your clothing, the walls of most rooms, most flooring, skin, and even paper are all rough when viewed at the microscopic level. The picture at the right depicts a highly magnified, microscopic view of the surface of a sheet of paper. Reflection off of smooth surfaces such as mirrors or a calm body of water leads to a type of reflection known as specular reflection. Reflection off of rough surfaces such as clothing, paper, and the asphalt roadway leads to a type of reflection known as diffuse reflection. Whether the surface is microscopically rough or smooth has a tremendous impact upon the subsequent reflection of a beam of light. The diagram below depicts two beams of light incident upon a rough and a smooth surface.



Specular Reflection (Smooth Surfaces) **Diffuse Reflection (Rough Surfaces)**

A light beam can be thought of as a bundle of individual light rays which are traveling parallel to each other. Each individual light ray of the bundle follows the law of reflection. If the bundle of light rays is incident upon a smooth surface, then the light rays reflect and remain concentrated in a bundle upon leaving the surface. On the other hand, if the surface is microscopically rough, the light rays will reflect and diffuse in many different directions.

Phong Illumination Model :

Phong shading refers to a set of techniques in 3D computer graphics. Phong shading combines a model for the reflection of light from surfaces with a compatible method of estimating pixel colors using interpolation of surface normals across rasterized polygons. The model of reflection may also be referred to as the Phong reflection model, Phong illumination or Phong lighting. It may be called Phong shading in the context of pixel shaders or other places where a lighting calculation can be referred to as "shading". The interpolation method may also be called Phong interpolation, and is also usually what is referred to by "per-pixel lighting". Typically it is called "shading" when being contrasted with other interpolation methods such as Guard shading or flat shading. The Phong reflection model may be used in conjunction with any of these interpolation methods. Phong reflection is an empirical model of local illumination. It describes the way a surface is lit as a combination of the diffuse lighting of rough surfaces with the specular reflection of shiny surfaces. It is based on But Tuong Phong's informal observation that for very shiny surfaces the specular highlight was small and the intensity fell off rapidly, while for duller surfaces it was larger and fell off more slowly. Additionally it includes an ambient term to simulate the way that a low level of light is typically present everywhere in a scene.

Q. 8. (a) What is an image? How quality of an image can be improved with filtering?

Ans. Image :

A visual representation (of an object or scene or person or abstraction) produced on a surface; "they showed us the pictures of their wedding"; "a movie is a series of images projected so rapidly that the eye integrates them" Image compression is minimizing the size in bytes of a graphics file without degrading the quality of the image to an unaccceptable level. The reduction in file size allows more images to be stored in a given amount of disk or memory space. It also reduces the time required for images to be sent over the Internet or downloaded from Web pages. There are several different ways in which image files can be compressed. For

Internet use, the two most common compressed graphic image formats are the JPEG format and the GIF format. The JPEG method is more often used for photographs, while the GIF method is commonly used for line art and other images in which geometric shapes are relatively simple.

Other techniques for image compression include the use of fractals and wavelets. These methods have not gained widespread acceptance for use on the Internet as of this writing. However, both methods offer promise because they offer higher compression ratios than the JPEG or GIF methods for some types of images. Another new method that may in time replace the GIF format is the PNG format.

A text file or program can be compressed without the introduction of errors, but only up to a certain extent. This is called lossless compression. Beyond this point, errors are introduced. In text and program files, it is crucial that compression be loss less because a single error can seriously damage the meaning of a text file, or cause a program not to run. In image compression, a small loss in quality is usually not noticeable. There is no "critical point" up to which compression works perfectly, but beyond which it becomes impossible. When there is some tolerance for loss, the compression factor can be greater than it can when there is no loss tolerance. For this reason, graphic images can be compressed more than text files or programs.

Q. 8. (b) Write and explain the flood fill algorithm.

Ans. Flood fill algorithm :

A recursive graphics algorithm used to fill in irregular-shaped regions with a solid color.

- We often want to "color" regions which are defined by an irregular-shaped boundary.
- Our earlier "scan line-fill" algorithm worked well for regions with boundaries defined by straight line-segments, or by circular arcs.
- But more complicated regions could lead to programming difficulties if the scanline approach is taken.
- A "recursive" approach may be simpler.
- It works by successively reducing a very complicated problem to one that is just a little bit easier to solve.
- We apply that idea to a graphics problem: coloring the interior of an irregular region .

Algorithm :

```
void fill (int x, inty, int interiorcolor, int newcolor)
    {
        if(get_pixel (x, y) = = interiorcolor)
        {
            put_pixel (x, y, newcolor); fill (x-1, y, interiorcolor, newcolor);
            fill (x+1, y, interiorcolor, newcolor); fill (x, y-1, interiorcolor, newcolor);
            fill (x, y+1, interiorcolor, newcolor);}}
```

- We need a way to create irregular regions.

- We will do it by using "bezier curves".
- We subdivide the viewpoint into quadrants.
- We form a bezier curve in each quadrant.
- These bezier curves will connect together.
- First and last control-points are mid-axes.
- Other control-points are randomly chosen.