17/3/10 – ~~15 18~~ 29, 32, 39, 42, 44, 47, 52, 53,
PR – 46, 47, 9, 52, 12, 15, 22, 16, 30, 8, 10, 31, 54, 28, 43, 8, 33, 60, 34, 36, 56,
45, 50, 48, 49, 51, 25, 38, 57, 17

4 Pages

TERNA PUBLIC CHARITABLE TRUST'S

TERNA POLYTECHNIC

Sector-1, Koparkhairane, Navi Mumbai - 400 709.

PROGRESSIVE THEORY TEST I / II / 20.....

Implicit & Explicit when fun^n. is Known

1

To be filled in by the candidate

Subject :

Sr. No: 31471

Course : CM / IF / EJ / EX

Year : I / II / III

Roll No.:

Date :

Signature of the Invigilator

Main Answer book	No. of Suppliments	Total
1		

Que No.	1	2	3	4	5	6	Total Marks	Sign. of Examiner
Marks								

(Please start writing from here)

2) Explicit
$y = f(x)$.
- Single valued
- Cannot represent vertical line

3) Parametric
$P(t) = f(t), g(t)$
$P(t) = x(t), y(t)$
$t = 0$ to 1.

Ch. 5. CURVES AND FRACTALS.

1) Implicit Curve :- Procedure can test pt on curve. $f(x, y) = 0$
Multiple y value for one x
multivalued curve eg Circle - $x^2 + y^2 - R^2 = 0$.

* Introduction :-

Objects in real world are not always made up of regular geometric shapes, which may involve curves. Curves cannot be represented by using exact mathematical functions or equations.

Drawing curves involves complex mathematical analysis in the form of various interpolation techniques by maintaining the continuity and other properties.

Geometrically, curves are classified into two broad categories —

i) Namable Curves — are those ~~curves~~ part of geometry that can be analysed mathematically by equations (Also called as curves from classical geometry)

2) Unnamable curves.

* Curve Generation :-

The curves can be drawn with two approaches as follows :-

1) First approach is to develop curve generation algorithm such as DDA. A true curve is generated with this approach.

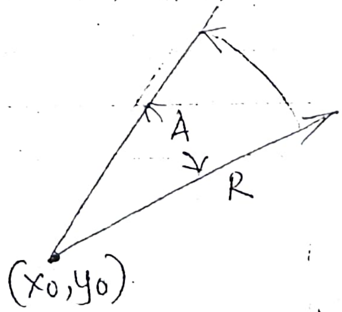
2) Second method is to approximate a curve by a number of small straight line segments. Interpolation technique is used for this approach.

* Arc Generation using DDA Algorithm :-

The differential equations for arc co-ordinates can be written in terms of one angle parameter A as follows :-

$$\left.\begin{aligned} x &= R\cos A + x_0 \\ y &= R\sin A + y_0 \end{aligned}\right\} \text{equation } 1$$

where (x_0, y_0) is center of curvature & R is radius of the arc.



Differentiating the above equations,

$$dx = -R \sin A \cdot dA$$
$$dy = R \cos A \cdot dA$$
} equation 2

From eqn 1, $R \cos A = x - x_0$
$R \sin A = y - y_0$

Substituting it in eqn. 2,

$$dx = -(y - y_0)\, dA$$
$$dy = (x - x_0)\, dA$$

Now, changes in dx and dy gives the amount to add to an old point on the arc to get new point.

$$\therefore x_2 = x_1 + dx$$
$$= x_1 - (y - y_0) \cdot dA$$
$$y_2 = y_1 + dy$$
$$= y_1 + (x - x_0) \cdot dA$$

These equations form the basis for an arc generation algorithm.

Algorithm ARC DDA :-

Step 1 :-
Accept center of curvature (x_0, y_0)
A - angle and (x, y) starting point

Step 2 :-
calculate dA

$dA = \min(0.1, 1/(3.2 * (|x-x_0|+|y-y_0|)))$

angle = 0.

Step 3 :-

Draw the first point.

putpixel (x, y, color)

Step 4 :

Increment the value of angle from 0 to A. if (angle < A)

~~for (angle = 0; angle < A; angle ++)~~

{

$x = x - (y_0 - y) * dA$

$y = y + (x - x_0) * dA$. $x = x - (y - y_0) * dA$

angle = angle + dA; $y = y - (x - x_0) * dA$

Set corresponding pixel.

putpixel (x, y, color)

}

Step 5 :-

End.

Advantages :-

1) Easy to implement

2) The differential equations for ellipse and circ are easy to implement than generation algorithms.

Disadvantages :-

1) Only the end points are not sufficient to specify the curve, so different file structure should be used.

4 Pages

TERNA PUBLIC CHARITABLE TRUST'S

TERNA POLYTECHNIC

Sector-1, Koparkhairane, Navi Mumbai - 400 709

PROGRESSIVE THEORY TEST I / II / 20.....

2

To be filled in by the candidate

Subject :

Sr. No. 31467

Course : CM / IE / EJ / EX

Year : I / II / III

Roll No. :

Date :

Main Answer book	No. of Suppliments	Total
1		

Signature of the Invigilator

Que No.	1	2	3	4	5	6	Total Marks	Sign. of Examiner
Marks								

(Please start writing from here)

3) Clipping algorithm only works for the points and straight lines.

~~* Interpolation :-~~

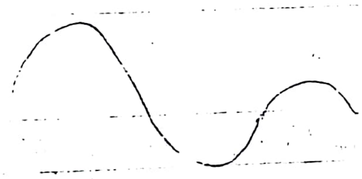
Problems in ~~True~~ Curve Generation Approach :-

1) For specifying a curve, we need more infⁿ than just its endpoints.
2) Difficult to apply transformation.
3) New clipping algorithm is required for clipping arcs.
4) Curve generation algorithms for curves other than circular or elliptical are complex.

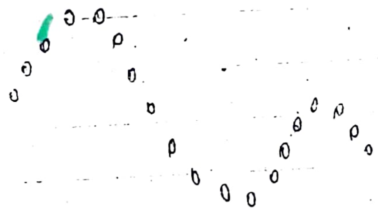
* Interpolation :-

— In real practice, we have to deal with some complex curves for which no direct mathematical funⁿ. is available

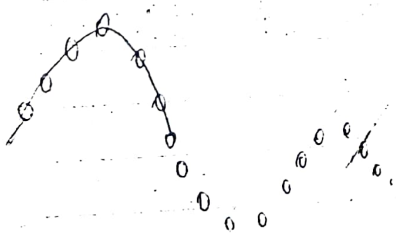
- Such curves can be designed using approximation methods.
- If we have set of sample points which lie on the specific curve, then we can draw the required curve by filling portions of curves with the pieces of known curves which pass through nearby sample points.
- The gap between the sample points can be filled by finding the co-ordinates of the points along the known approximating curve and connecting these points with line segments.



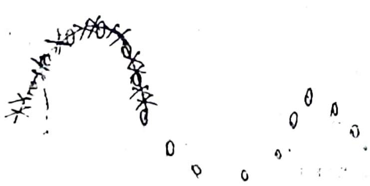
Unknown Curve



Known Sample points



Fit region with a known curve



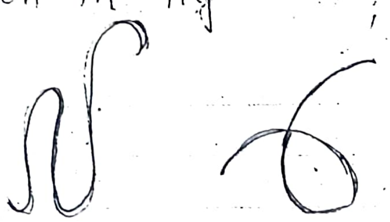
Calculate more points from Known Curve. Then, draw line connecting points.

- The main task in this process is to find suitable mathematical expression for the known curve.
- There are polynomial, trignometric, exponential and other classes of functions that can be used to approximate the curve.

- Usually polynomial functions in parametric form are preferred.
  Polynomial functions in parametric form can be given as,

$$x = f_x(u)$$
$$y = f_y(u)$$
$$z = f_z(u)$$

- Parametric form allows multiple values (several values of y & z for given x value)
- Due to these multiple values, curves can double back or even cross themselves, as shown in fig.



- Let: see the interpolation process,
  $(x_1, y_1, z_1), (x_2, y_2, z_2), \ldots (x_n, y_n, z_n)$
- Functions can be given as,

$$f_x(u) = \sum_{i=1}^{n} x_i \cdot B_i(u)$$

$$f_y(u) = \sum_{i=1}^{n} y_i \cdot B_i(u)$$

$f_x(u) = \sum_{i=1}^{n} x_i \cdot B_i(u)$

Def^n of $B_i(u)$

- Function $B_i(u)$ is called as blending function.
- For each value of parameter u, the blending function determines how much the ith sample points affects the position of curve.
- In other words, we can say that each sample points tries to pull the curves in its direction and the function $B_i(u)$ gives the strength of the pull.

Blending function which is 1, can be given as,

$$B_i(u) = \frac{(u+1)(u)(u+1) \ldots [u-(i-3)][u-(i-1)] \ldots [u-(i-2)]}{(i-1)(i-2)(i-3) \ldots (1)(-1) \ldots (1-r)}$$

Approximation of curve using above exp^n. is called lagrange interpolation.

Blending fun^n. for four sample points can be given as,

$$B_1(u) = \frac{u(u-1)(u-2)}{(-1)(-2)(-3)}$$

$$B_2(u) = \frac{(u+1)(u-1)(u-2)}{1(-1)(-2)}$$

$$B_3(u) = \frac{(u+1)u(u-2)}{(2)(1)(-1)}$$

$$B_4(u) = \frac{(u+1)u(u-1)}{(3)(2)(1)}$$

4 Pages

TERNA PUBLIC CHARITABLE TRUST'S

TERNA POLYTECHNIC

Sector-1, Koparkhairane, Navi Mumbai - 400 709.

PROGRESSIVE THEORY TEST I / II / 20.....

(3)

To be filled in by the candidate

Subject : Sr. No 31466

Course : CM / IF / EJ / EX

Year : I / II / III Roll No. :

Date :

Signature of the Invigilator

Main Answer book	No. of Suppliments	Total
1		

Que No.	1	2	3	4	5	6	Total Marks	Sign. of Examiner
Marks								

(Please start writing from here)

- Using above blending functions, expression for curve passing through sampling points can be realised as follows,

$$x = x_1 B_1(u) + x_2 B_2(u) + x_3 B_3(u) + x_4 B_4(u)$$
$$y = y_1 B_1(u) + y_2 B_2(u) + y_3 B_3(u) + y_4 B_4(u)$$
$$z = z_1 B_1(u) + z_2 B_2(u) + z_3 B_3(u) + z_4 B_4(u)$$

* Interpolating Algorithm :-

1) Get the sample points.
2) Get intermediate value of u to determine intermediate points.
3) Calculate blending function values for middle section of curve.
4) Calculate blending funn. values for first section of curve
5) Calculate blending funn. values for last section of curve.
6) Multiply sample point by blending funn. to give an approximation curve.

7) Connect neighbouring points using straight line segments.

8) Stop.

* B-Splines :-

— The Lagrange interpolation program is implemented by using blending function, in which, the sum of blending functions is not 1 at every value of u. The blending function is basically designed to sum 1 for integer values of u, not for the fractional values. We will get the flat behaviour. This problem can be solved by dividing the blending function, calculated by their sum for each value of u, normalizing the functions.

— One more problem is, each section of the curve is connected to the next section at a sample point. Meaning that there can be corners at the sample points and may not have a completely smooth curve.

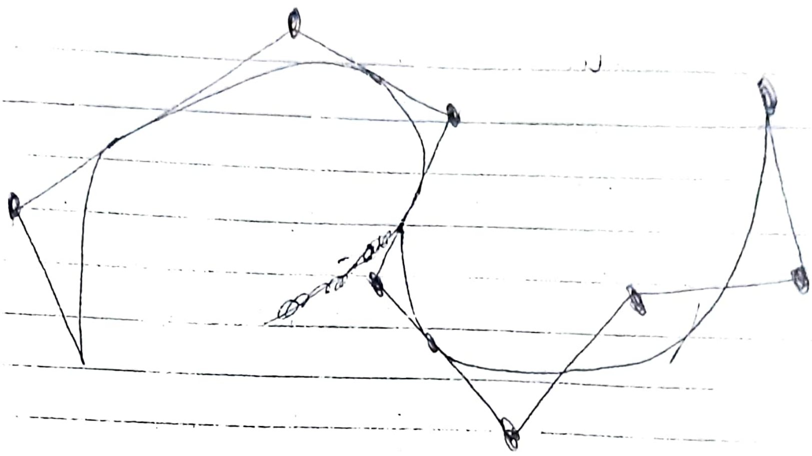
— Instead of forcing the curve to pass through the sample point, gently pull the curve into the neighbourhood of the sample point.

— The set of blending functions which take this approach and also always sum to 1 are called B-Splines ~~curves~~.

Advantages of B-Splines:~~Ex~~

① Degree of B-Spline polynomial can be set independently of no. of control points.

② B-Spline allow local control over shape of spline curve of surface.

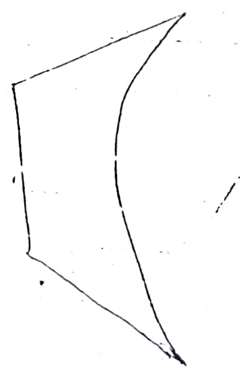


* Bezier Curve :-

The cubic Bezier curve is adequate for most graphics applications. This curve requires four control points. These four points completely specifies the curve.

8 We cannot extend Bezier Curve, but we can take four more points and we can construct a second Bezier Curve, which can be attached to the first.

Bezier Curve & its control points are as shown :-

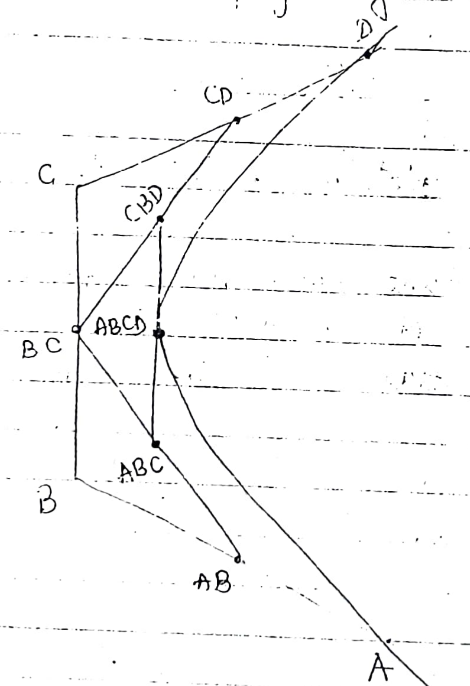


To connect two Bezier Curve, just make the first control point of second curve match the last control point of first curve

Equations describing the Bezier Curves are as follows:-

$$x = x_4 u^3 + 3x u^2(1-u) + 3x_2 u(1-u)^2 + x_1(1-u)^3$$
$$y = y_4 u^3 + 3y u^2(1-u) + 3y_2 u(1-u)^2 + y_1(1-u)^3$$
$$z = z_4 u^3 + 3z u^2(1-u) + 3z_2 u(1-u)^2 + z_1(1-u)^3$$

Bezier curve can be constructed without any reference to the above equations. Actually, the Bezier curve can be constructed simply by taking midpoints.



Advantages of Bezier curve.

I Bezier curve satisfies the convex hull property, means that the curve lies within the boundary stretched over all four of the control points, which is useful in clipping.

4

2) It can be extended to surfaces.
3). The bicubic Bezier curve can be constructed. 4x4 control points.

Properties of Bezier curve :-

1) Curve always passes through the first and last control points.
i.e. $P(0) = P_0$
$P(1) = P_n$.

2). Multiple values ⇒ The parametric formulation of Bezier curve allows it to represent multiple valued shapes. If the first & last control points coincide the curve is closed.

3) A Bezier curve is independent of the co-ordinate system used to measure the location of control points.

4) Bezier curve do not provide localized control. moving any control points will change the shape of curve.

5) Bezier curve lies within the convex polygon boundary of the control pts.

6. Bezier curve can be pieced together to describe a more complex curve.

~~Topo~~

* Fractals :

objects in nature are of rough, jagged, random edges. Mountains, trees rivers can be drawn with a lot of specification. This can be done with the help of fractals. In which machine can draw jugged lines provided two endpoints. In the fractal - geometry method, procedure rather than equation are used to model object.

A fractal object has two characteristics:

1) infinite detail at every point.
2) Certain self similarity between the object part and overall feature of object.

Types
1) Self-Similar
- Trees
- plants
2) Self-Affine
S_x, S_y, S_z
water, cloud, Terrain
3) Invariant
non-linear transfn.

* fractal Dimension :-

1) An Object which is composed of elast or clay can be deformed into line or line segment it's dimension $D_t = 1$.

If the object is deformed into plane or half plane of disk its dimension $D_t = 2$ and if it deforms into all space or into ball it's dimension $D_t = 3$, where D_t = topological Dimension.

2) A line segment of length L & dividing it into N identical pieces each of length $l = L/N$. The piece will look like original one, can only scaled $1/s = l/L$. To assemble the original line segments from segment scaled by $1/s$, add $N = s^1$.

$d = L/N$ ①

$s = \frac{L}{l}$ ②

$N = s$ ie. $N = s^1$.

If we scale a line segment by factor $\frac{1}{s}$, then we have to add N pieces together to get original line segments.

4 Pages

TERNA PUBLIC CHARITABLE TRUST'S

TERNA POLYTECHNIC

Sector-1, Koparkhairane, Navi Mumbai - 400 709.

PROGRESSIVE THEORY TEST I / II / 20.....

To be filled in by the candidate

Subject :

Sr. No. 31165

Course : CM / IF / EJ / EX

Year : I / II / III

Roll No.:

Date :

Signature of the Invigilator

Main Answer book	No. of Supplements	Total
1		

Que No.	1	2	3	4	5	6	Total Marks	Sign. of Examiner
Marks								

(Please start writing from here)

1) Scaling square down by 1/S small square will be obtain
to build original square from small square $N = S^2$ will be required

2) example ⇒ consider a square
if S = 2 it takes 4 square $N = S^2$
if S = 3 it takes 9 square $N = S^3$

4) for cube The exponent used in points 1, 2, 3 is measuring the dimension.

5) If we scale an object, then the dimension D of an object is given by relation

$$N = S^D$$

so D is fractal dimension.

$$D = \log N / \log S$$

When FD is closer to 1, more smooth

$N = 8^1$

$2 = 8^1 \cdot 8 = 2$

$N = S^2$

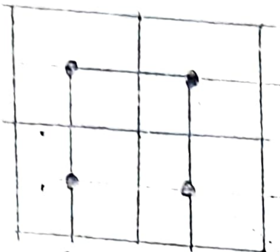
$4 = S^2$

$9 = S^D$

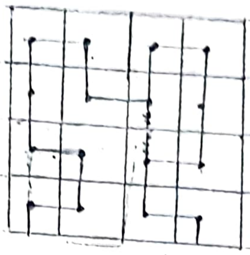
* Hilbert's Curve :-

1) this curve can be built by the following successive approximation

2) It is begin with square. the first approx^n will be divide the square into 4 quadrants and Draw the curve which connect centre points of each.



first approximation



Second Approximation

In the second approximation further divide each of the quadrants and connect the centres of each of these finer division before moving to the next major quadrants.

The fractal dimension can be determined as at each subdivision the scale is changed by 2 but the length is changed by 4. for square it takes 4 curves of the half scaled to build the full sized object so dimension D can be given as

$$4 = 2^D$$

$D = 2$

The Hillbert curve has topological dimension 1 But fractal dimension 2.

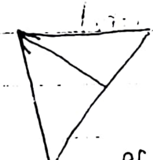
$4 = 2^D$

1) Curve never crosses itself.
2) Curve is close to every pt in square
3) Curve - 2 as fine with each subdivision.
4) No limit to subdivisions
5) length of curve is changed by 4
6) Scale

* fractal Lines

Computer Procedure can terminate when length become less than a pixel & provide the computer's best approximation to the fractal.

fractal lines



17

offset the Halfway point

The halfway point for fractal line, if we had straight - line betⁿ (x_1, y_1, z_1) & (x_2, y_2, z_2) the halfway pt. would be mid-pt:

$[(x_1+x_2)/2, (y_1+y_2)/2, (z_1+z_2)/2]$

But fractal line, add offset term to each co-ordinate.

$[(x_1+x_2)/2 + dx, (y_1+y_2)/2 + dy, (z_1+z_2)/2 + dz]$

To get random effect, the amount of the offset cal. as.

$dx = L * w * Gauss$

Where L = length of segment.

W = weighing funⁿ governing the curve roughness

Guass = It return a Guassian variable with 0 mean.

The fractal-line algorithm draws a fractal line from the current position to a specified point. It takes endpoints, weight factor and recursive depth as arguments.

Guass

Advantages :- All the clipping & transformation (CT) applied.

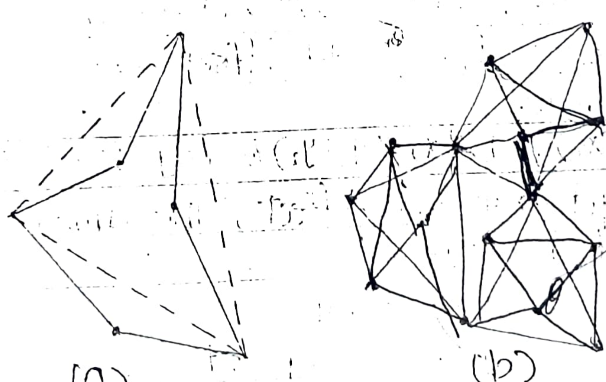
DisAdvantage :- A large display file is needed to hold the tiny segments and time must seperate of processing each of them.

fractal surfaces :-

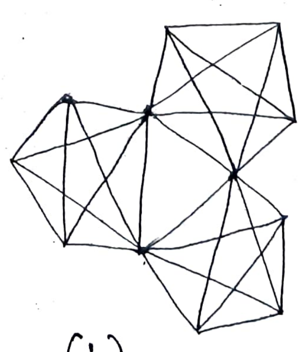
To draw a 3-dimensional mountain range, fractal surfaces are used.

Consider a traingle given three vertex points in space generate the fractal surface for the area between them. Consider each edge of the traingle, a fractal line along each edge and compute its half way point.

By connecting these halfway points with line segment, we can subdivide the surface into four smaller traingles. then we can recursively apply the method to each of the small traingle to subdivid the surface.



(a) (b)



(b)

4 Pages

TERNA PUBLIC CHARITABLE TRUST'S

TERNA POLYTECHNIC

Sector-1, Koparkhairane, Navi Mumbai - 400 709.

PROGRESSIVE THEORY TEST I / II / 20..... (S)

To be filled in by the candidate

Subject : Sr. No. 31464

Course : CM / IF / EJ / EX

Year : I / II / III Roll No. :

Date : Signature of the Invigilator

Main Answer Book	No. of Suppliments	Total
1		

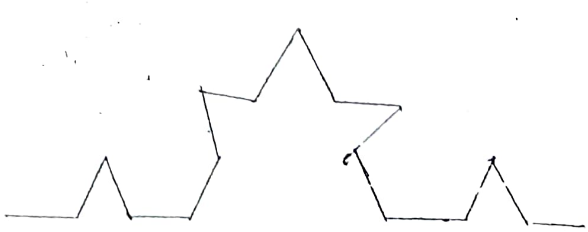
Que No	1	2	3	4	5	6	Total Marks	Sign. of Examiner
Marks								

(Please start writing from here)

* Koch curve :-

In koch curve, begin at a line segment. Divide it into 4 equal parts with scaling factor 1/3 and adjust two middle segment to form two adjacent sides of an equilateral traingle.

This will give the curve which begins and ends at the same place as the original segment but is built of 4 equal length segment, with each 1/3rd of the original length. so the new curve has 4/3 the length of original segments. 4/3.



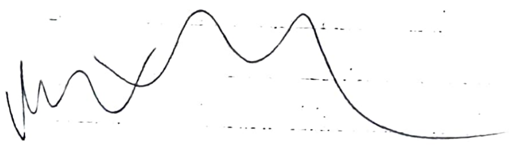
The curve has gained more wiggles and its length is now 16/9 times the original.

Since each repetition increases the length by a factor of 4/3.

It's topological dimension is 1, its fractional dimension is $4 = 3^D$. Solving this fractal dimension gives.

$$D = \log 4 / \log 3 = 1.2618$$

The point sets, curves, surfaces gives a fractal dimension that are greter than the topological dimension are called fractals.



4 Pages

TERNA PUBLIC CHARITABLE TRUST'S

TERNA POLYTECHNIC

Sector-1, Koparkhairane, Navi Mumbai - 400 709.

PROGRESSIVE THEORY TEST I / II / 20.....

To be filled in by the candidate

Subject :

Course : CM / IF / EJ / EX

Year : I / II / III

Date

Sr. No. 31674

Roll No. :

Signature of the Invigilator

Main Answer book	No. of Suppliments	Total
1		

Que No.	1	2	3	4	5	6	Total Marks	Sign. of Examiner
Marks								

(Please start writing from here)

Ch. 6 RASTER GRAPHICS AND INTERACTIVE GRAPHICS.

1) * Random Scan Display :-

Advantages :-

1) It has very high resolution and limited only by the monitor.
2) Easy animations are possible, just draw at different positions.
3) It requires little memory.

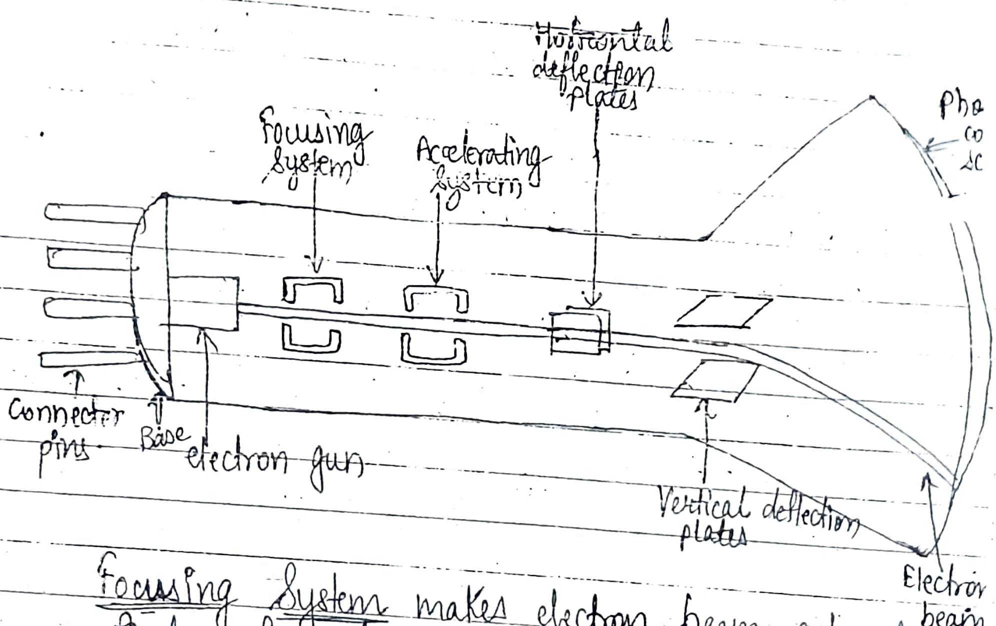
Disadvantages :-

1) It requires intelligent electron beam i.e. processor controlled beam.
2) It cant draw a complex image as it has limited display.
3) It is very expensive in terms of colour.

2) Cathode Ray Tube :-

CRT is an evacuated glass tube which contains electron gun, Focusing System, Accelerating System, Deflection system and Screen with phosphor coated inside.

Electron gun :- Generates beam of electrons which passes through focusing and deflect system and strikes on specified positions on phosphor coated screen. When electron beam strikes phosphor that phosphor emits light & thus point on screen is visible.



Focusing System makes electron beam extremely finely focused and makes highly concentrated small spot when it strikes phosphor. Otherwise beam will spread when it reaches the screen.

Accelerating System :- consists of two metal plates mounted perpendicular to beam axis to accelerate beam to necessary velocity.

1) Raster Scan Display Component :-

Host CPU → Display Processor → Frame Buffer → Display Controller → Visual Display Unit

Frame buffers used for raster display :-

1) Rotating memory frame buffer.
2) Shift Register frame buffer.

Frame buffer used for random display :-

1) Random access frame buffer.
2) Multiple plane. — " —

6) OpenGL :-

Definition :- OpenGL is a software interface to the graphics hardware.

1) → This interface consists of about 150 distinct comm that can be used to specify objects & operations needed to produce interactive 3D applications.

2) → Main purpose of OpenGL is to render 2D & 3D objects in frame buffer.

3) → OpenGL performs several processes on this data to convert it to pixels to form final desired image in the framebuffer.

4) → OpenGL is a Configurable State Machine. Its i/ is 2D or 3D data and its o/p is framebuffer.

5) → OpenGL is designed as streamlined, h/w-indepe interface to be implemented on different h/w platforms. There are no commands for this.
→ Instead, you must work through whatever windowing ~~system controls~~ particular hardwa you are using.

6) → OpenGL does not provide high-level comman

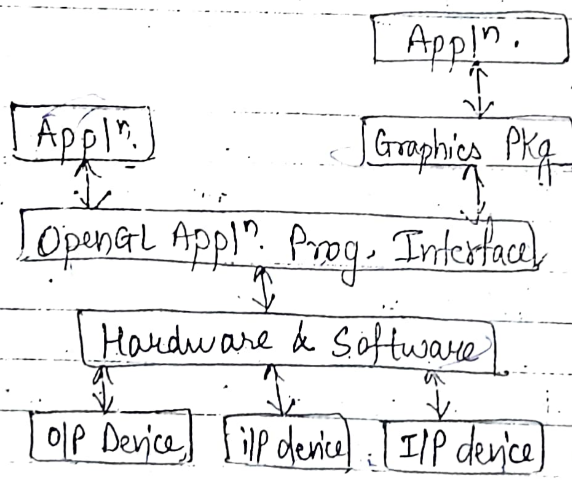
OpenGL is not used for :-

12 1) It does not provides means of modeling complex objects
2) It does not provide support for peripherals
3) It does not provide windowing or GUI.

7) → With OpenGL, you must build up your desired model from small set of geometric primitives — points, lines & polygons.

OpenGL Used for :- (Applications)

1) Real time applications.
12 2) Fast preview for visualization.
3) Interactive virtual environments.
4) Developing Video Games.



You must include gl.h header file.
All appl^n. use GLU (Graphics Library Utility).

#include <GL/gl.h>

#include <GL/glut.h>

To draw a primitive :-

1) Call glBegin() to start.
2) Call primitive with list of vertices & their attributes
3) Call glEnd() to end.

OpenGL Primitives :-

Points, Lines, Polygon, Triangle, Quad, Quad Strip, triangle strip, triangle fan.

4 Pages

TERNA PUBLIC CHARITABLE TRUST'S

TERNA POLYTECHNIC

Sector-1, Koparkhairane, Navi Mumbai - 400 709.

PROGRESSIVE THEORY TEST I / II / 20.....

(2)

To be filled in by the candidate

Sr. No. 31675

Subject :

Course : CM / IF / EJ / EX

Year : I / II / III

Roll No.:

Date :

Signature of the Invigilator

Main Answer book	No. of Supplements	Total
1		

Que No.	1	2	3	4	5	6	Total Marks	Sign. of Examiner
Marks								

(Please start writing from here)

3) Shadow – Mask CRT :-

5) Raster Scan Monitors :-

Raster displays also known as bit-mapped or raster display.

1) Rasters :-

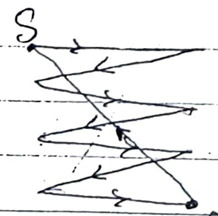
Raster is series of adjacent parallel lines which together from ~~raster~~ image on display screen.

Pixel positions have x, y co-ord.

2) Pixel Principles :-

Entire collection is called as Raster. Each row of pixel is called Raster line or Scan line.

3) Raster Scan :-



→ This technology is very cost effective, inexpensive and because of availability of large memory its refresh speed is high.

→ Disadv. of raster scan is jagged nature of line.

Now-a-days electron gun is built to contain focusing system and accelerating system within same unit.

The position of dot on screen can be controlled by deflection electron beam. ~~is done~~

3) Storage type Output devices :-

a) DVST

b) Plasma Display Panel.

c) Laser Scan display :-

→ Image is created with laser tracking dark dots and lines on a photochromatic film, which are projected onto the large screen.

→ It produces high resolution image (100 dot per inch) & permanent in the sense that fresh film is required for different Image.

Process

→ 1) In laser scan display, laser is deflected by pair of tiny mirrors to trace out the desired image on sheet of photochromatic film

→ 2) The extremely small mirrors used for deflecting the laser are controlled by the electrical signals recieved by display controller

→ 3) Light from laser leaves dark spot on transparent film & Image is deposited on scre

→ 4) light projection system is used to project image onto screen, which may be as large as 3x4 feet.

4) Refresh Type Output Devices :-

1) Standard CRT :-

When electron beam stops, it moves during the scanning sequence or at end of session, the fades into an after glow, called as phosphorescence. The time taken for this to fade out is known as persistance.

For continous viewing, image is refreshed i.e. repeatedly produced at intervals so that the changing image appears continously. The screen refresh cycle speed is of 30 times per second, then screen is completely refreshed.

2) Beam Penetration CRT :-

CRT with single phosphor can generate only one color.

A colour CRT display uses multiple ~~colours~~ layer phosphor.

The screen is coated with a layer of green phosphor over which a layer of red phosphor is deposited.

When low potential electron beam strikes screen only red phosphor is excited, thus producing red trace. Higher velocity beam can penetrate green phosphor increasing green component of light output.

It can also produce limited range of colours such as orange, ~~gre~~ yellow, etc.

Advantages :-

1) Multiple colours are possible.

Disadvantages :-

1) Limited range of colours are possible.
2) Hardware & software must be designed to introduce adequate delays between colour ~~sa~~ changes.

(3) 4 Pages

TERNA PUBLIC CHARITABLE TRUST'S

TERNA POLYTECHNIC

Sector-1, Koparkhairane, Navi Mumbai - 400 709.

PROGRESSIVE THEORY TEST I / II / 20.....

To be filled in by the candidate

Subject : Sr. No. 31676

Course : CM / IF / EJ / EX

Year : I / II / III Roll No.:

Date : Signature of the Invigilator

Main Answer book	No. of Supplimonts	Total
1		

Que No.	1	2	3	4	5	6	Total Marks	Sign. of Examiner
Marks								

(Please start writing from here)

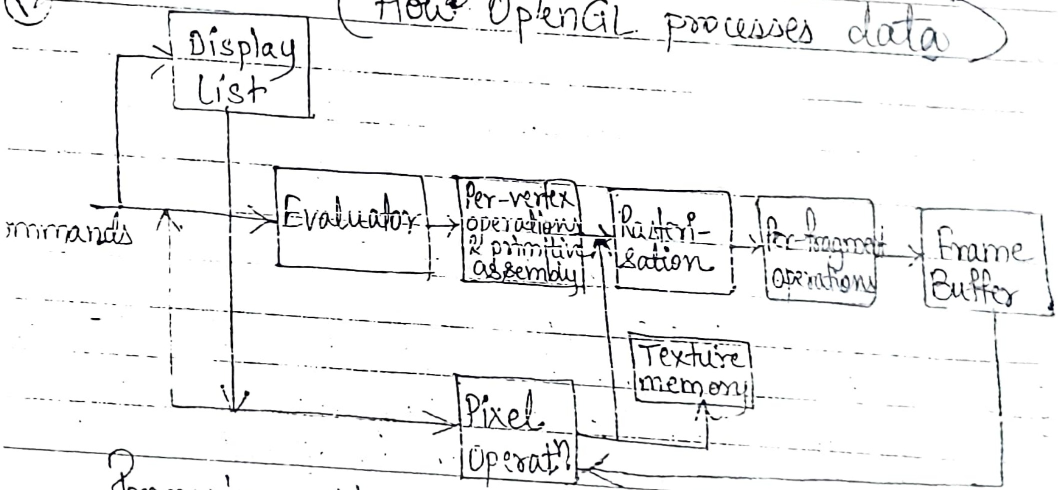
* Main features of OpenGL :-

16 OpenGL implements :-

1) Geometric Primitives (Mathematical Description of obj...
2) Colour Coding in RGBA (Red-Green-Blue, Alpha)
3) Viewing & Modeling (Arranging objects in 3D scene, move camera & select desired vie...
4) Texture mapping → To bring realism.
5) Materials lighting.
6) Double Buffering → Helps to eliminate flickering from animations.
7) Anti-aliasing.
8) Gouraud Shading → apply smooth shading to 3D object.
9) Z-buffering → keep track of proximity of viewer object.
10) Atmospheric effects → like fog, smoke ~~has~~ make images more realistic.
11) Alpha blending → eg. transparent light blue window to dra... in front of red bo...
12) Stencil planes → Restrict drawing to certain portions of screen.
13) Display list permits storage of commands in list for later rendering.

14) Feedback, Selection & Picking.
15) Raster primitives.
16) Pixel Operations.
17) Transformations in 3D, etc.

* Basic OpenGL Operations :-
(How OpenGL processes data)



Processing stages are as follows :-

Display list :-
Rather than having all commands proceed immediately through pipeline, you can choose to accumulate some of them in display list for processing later.

Evaluator :-
Evaluator stage of processing provides an efficient way to appropriate curve and surface geometry by evaluating polynomial commands of input values.

Per-vertex operations & primitive assembly :-
OpenGL processes geometric primitives all of which are described by vertices.

Vertices are transformed & lit, and primitives are clipped to viewport in preparation for rasterization.

Rasterisation :-

Produces series of frame buffer addresses & associated values using 2D description of pt, line or polygon. Each fragment produced is fed into last stage, Per-fragment operations.

Per-fragment operations :-

Final operation performed on data stored as pixels in frame buffer.

This includes conditional updates to frame buffer depending on various logical operations on pixel values.

Data can be input in form of pixels rather than vertices. Data in form of pixels such as might describe image for use in texture mapping, skips first stage & instead is processed in pixel operations stage. Pixel data is,

a) either stored as texture memory for use in rasterisation stage OR

b) Rasterized with resulting fragments mapped in frame buffer just as if they were generated from geometric data.

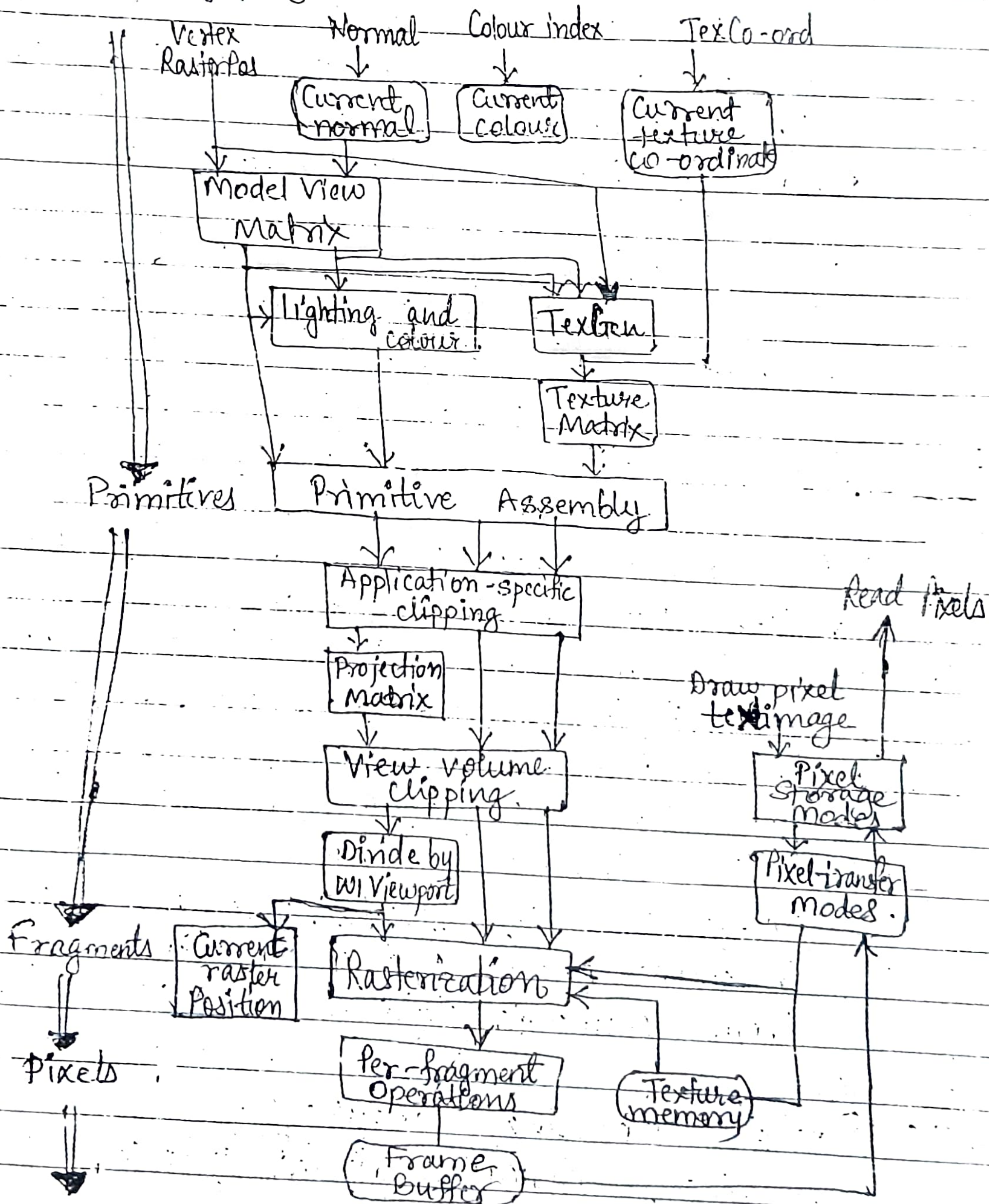
* OpenGL Processing Pipeline :-

Processing is basically divided into four sections.

1) Vertices
2) Primitives
3) Fragments
4) Pixels.

i) Vertices :-

This section relates the OpenGL commands that perform per-vertex operations to the processing stages.



(4)

4 Pages

TERNA PUBLIC CHARITABLE TRUST'S

TERNA POLYTECHNIC

Sector-1, Koparkhairane, Navi Mumbai - 400 709.

PROGRESSIVE THEORY TEST I / II / 20.....

To be filled in by the candidate

Subject :

Sr. No. 31677

Course : CM / IF / EJ / EX

Year : I / II / III

Roll No. :

Date :

Signature of the Invigilator

Main Answer book	No. of Supplimenis	Total
1		

Que No.	1	2	3	4	5	6	Total Marks	Sign. of Examiner
Marks								

(Please start writing from here)

Input Data :-

Various types of input data to OpenGL Pipeline :-

i) Matrix Transformations :-

Vertices and normals are transformed by modelview and projection matrices before they're used to produce an in in frame buffer.

ii) lighting and Colouring :-

Define desired lighting conditions & desired mater properties. Related commands use to control how lighti calculations are performed.

iii) Generating Texture Co-ordinates :-

Rather than explicitly supplying texture co-ordinate OpenGL can generate them as function of other vertex data. Then they are transformed by ~~tran~~ texture mat.

2) Primitives :-

During next stage of processing, primitives are converted to pixel fragments in several steps Primitives are clipped appropriately, whatever corresponding adjustments made to colour & texture data and co-ord. are transformed to window co-ord

Comp No. 409859700.

Finally, rasterisation converts clipped primitives to pixel fragments.

Primitive Assembly :-

Vertices are assembled into primitives with relevant edge flag, colour & texture infn. for each vertex.

Transforming to window coord :-

First, co-ord. are normalised & then converted to window co-ord.

Rasterization :- Process by which a primitive is converted to 2D image.

3. Fragments :-

OpenGL allows fragment produced by rasterisat. to modify corresponding pixel in frame buffer only if it passes series of tests. If it does pass, fragments data can be used directly to replace the existing frame buffer values, or it can be combined with existing data in the frame buffer, depending on state of certain modes.

Pixel Ownership Test :-

First test is to determine whether pixel correspondi to particular fragment is owned by current OpenGL context. If so, fragment proceeds to next test. If not, window system determines whether fragmen is discarded or whether any further fragment operatn. will be performed with that fragment. This test allows window system to control OpenGL behaviour.

4 Pages

TERNA PUBLIC CHARITABLE TRUST'S

TERNA POLYTECHNIC

Sector-1, Koparkhairane, Navi Mumbai - 400 709.

PROGRESSIVE THEORY TEST I / II / 20.....

5

To be filled in by the candidate

Sr. No. 31678

Subject :

Course : CM / IF / EJ / EX

Roll No.:

Year : I / II / III

Signature of the Invigilator

Date :

Main Answer book	No. of Suppliments	Total
1		

Que No.	1	2	3	4	5	6	Total Marks	Sign. of Examiner
Marks								

(Please start writing from here)

3) View every user warning & error dialogue.
4) Provide adequate user feedback.
5) Use sound, colour, animation & multimedia clip sparingly.
6) Help users to customize & preserve their prefer. work environment.
7) Design your interface so that users can do their task while being minimally aware of intert itself.

NEED FOR GRAPHIC STANDARDS :-

A standard architecture is needed for writing graphics applications & having them communicate with huge diversity of available h/w.
From ~~it~~ this, it can be seen:

19 a) why better graphics have not been available for personal computers.

b) why stronger efforts have not been made to ~~summarize~~ standardize production of grap. programs

To assist these problems, ANSI has formed technical committee to develop standards.

1.

- Graphical standards are provided for [illegible] transfer of graphics from one platform to another, which provided portability.

The first Standard accepted was GKS (Graphical Kernel System) & was adopted by ISO. GKS defines graphics functions at programmer level, specification of how those functions are assessed through high-level programming languages.

Other major graphical Standards available are :-

1) CORE
2) Programmer's Hierarchical Interactive GS (PHIGS)
3) Initial Graphics Exchange Standard (IGES)
4) Computer Graphics Metafile Standard (CGM)
5) Virtual Device Metafile (VDM).
6) Virtual Device Interface (VDI).

1) GKS :- Also called as 2D version of CORE

- Defines Common Interface to interactive computer graphics for creating, manipulating & displaying or printing computer graphics on diff. types of graphical o/p devices.
- First Standard for low-level CG introduced in 1977.
- Full Standard provides functional specifications for some 200 subroutines which perform graphics i/p & o/p in device independent way.
- Portable
- Purpose of GKS is :

a) For providing portability of graphics appl^n. programs.

b) To assist in understanding of graphics method by appl^n. program.