Polygon Surfaces

- In Graphics a 3D-object can be represented as a set of surface polygons.
- And all surfaces are described with a linear equation.
- To get a realistic picture, ~~inter~~ interpolating shading patterns are produced across the polygon surface to elliminate the polygon edge boundaries.

(To get a smoother surface we have to interpolate the surface)

Polygon Table

- The polygon surface can be specified with a set of vertex co-ordinate and associated attribute parametres.
- A polygon table is maitained for ~~giving~~ keeping information for polygon surfaces which can be processed for displaying and manupulating polygon surfaces.
- The polygon table ~~are~~ is catagorized into 2 parts:
  - (a) Geometric table
  - (b) Attribute table

Geometric table

→ It contains the vertex co-ordinate and parametre to identify the spatial orientation of the polygon ~~table~~ surface

Attribute Table

→ It contains the data regarding the degree of transparency of the object and it's surface reflectivity and texture characteri-stics.

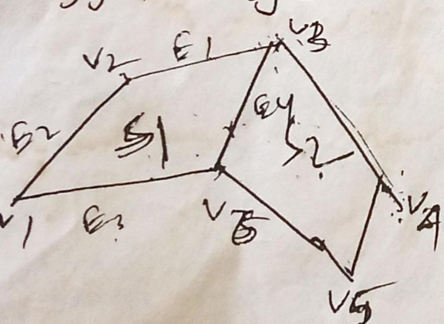
∴ Now to store the geometric data we have to create 3-tables ⇒

① ↳ **Vertex table**

* It contains the co-ordinate value for each vertex

② ↳ **Edge table**

* It contains the vertices for each polygon edge



V	
V1	x1 y1 z1
"	

E	V2 V3
E	

③ ↳ **Polygon surface table**

It contains the information regarding edges for each polygon ~~edges~~ surface

S1	E1, E2, E3, E4
S2	E6, E2, E6, E4

So the data from the edge table ~~is~~ can be used to draw component lines. So to draw ~~to~~ the object efficiently, the surface table has a back pointer to edge table & Edge table also has a back pointer to vertex table.

But to find out the common edge, the edge table has a forward pointer to polygon surface table

The additional geometric information which are to be stored are ~~as~~ slope for each edge, then ~~the~~ minimum & maximum values (x, y, z) for individual polygons

* The test performed by the graphics package *

1) Every vertex ~~is~~ is ~~at least~~ listed as an end point for atleast 2 edges.
2) Every edge is atleast a part of one polygon.
3) Every polygon is closed.
4) Each polygon has atleast one shared edge.
5) Each edge table and polygon surface table has a back and forward pointer.

Plane equation

17/10/06

To represent an object on o/p device we render go a sequence of transformation and surface rendering/covering procedures. and also the spatial orientation of the individual surface components of the objects. This inforⁿ can be obtained by the vertex co-ordinate value and ~~o~~ the equation of the polygon plane.

So the equation of the plane surface can be expressed as:

$$Ax + By + Cz + D = 0$$

where (x, y, z) is any point on the plane and A, B, C, D are constants describing the spatial property of the plane.

So the value of the A, B, C, D can be obtained by cramer's Rule:

$$A = \begin{bmatrix} 1 & y_1 & z_1 \\ 1 & y_2 & z_2 \\ 1 & y_3 & z_3 \end{bmatrix} \quad B = \begin{bmatrix} x_1 & 1 & z_1 \\ x_2 & 1 & z_2 \\ x_3 & 1 & z_3 \end{bmatrix}.$$

$$C = \begin{bmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{bmatrix} \quad \& \quad D = -\begin{bmatrix} x_1 & y_1 & z_1 \\ x_2 & y_2 & z_2 \\ x_3 & y_3 & z_3 \end{bmatrix}$$

For any point (x, y, z) which is not on the plane with the parameters A, B, C, D must satisfy the equation.

$$Ax + By + Cz + D \neq 0$$

a) If $Ax + By + Cz + D < 0$, the point (x, y, z) is inside the surface

b) If $Ax + By + Cz + D > 0$, the point (x, y, z) is outside the surface.

This inequality tests are valid in a right handed cartesian system and the

Curved lines AND surfaces

Curve and surface equations are expressed in either a parametric or non-parametric forms. But for graphics parametric representation are generally more convenient.

Quadric surface

→ Quadric surface are surfaces are described by the 2nd degree eqⁿ.

→ These surfaces includes, ~~~~ spheres, ellipsoids, tori, paraboloids, and hyperboloids.

Quadric surfaces, particularly ~~such~~ spheres and ellipsoids are common elements of graphics scenes. from which complex objects can be constructed.

Sphere

In cartesian co-ordinates, a spherical surface with radius r centered on the co-ordinate origin is defined as the set of points (x, y, z), that satisfy the equation,

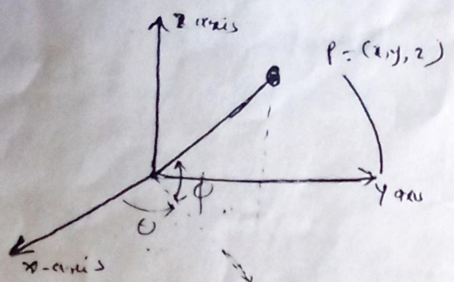
$$x^2+y^2+z^2=r^2 \quad \text{——} (1)$$

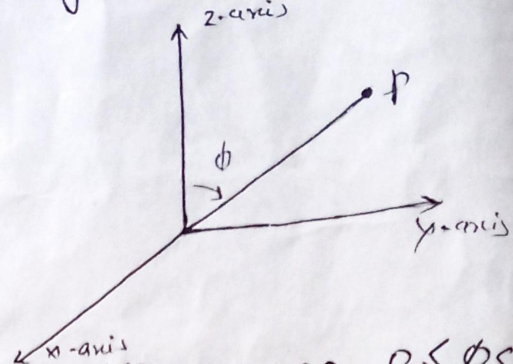
The spherical surface in parametric form using latitude and longitude angles is :-

$$\left.\begin{aligned} x &= r\cos\phi\cos\theta, \quad -\pi/2 \le \phi \le \pi/2 \\ y &= r\cos\phi\sin\theta \quad\quad -\pi \le \theta \le \pi \\ z &= r\sin\phi \end{aligned}\right\} \text{——} (2)$$

The eq^n 2 gives a symmetric range for the angular parametre θ and ϕ.

→ We can write the parametric spherical equations using standard spherical co-ordinates, where angle ϕ is specified as the colatitude





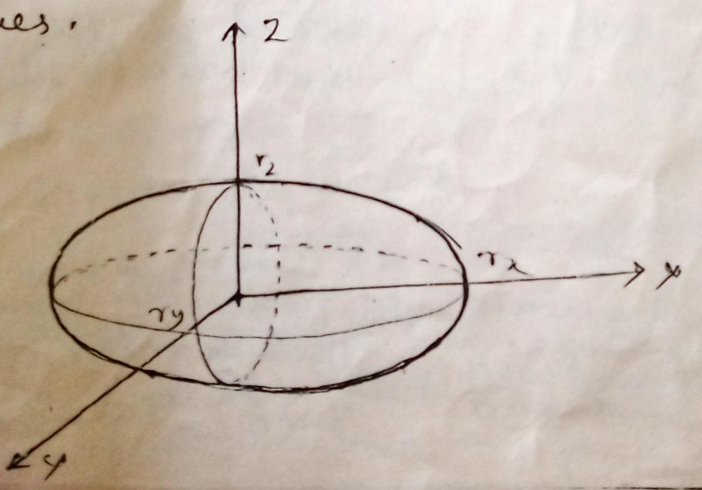
→ ϕ is defined over the range $0 \leq \phi \leq \pi$ and θ is taken in the range $0 \leq \theta \leq 2\pi$

→ We can set up the representation using parameters "u" and "v" over the range from 0 to 1 by substituting

$\phi = \pi u$ and $\theta = 2\pi v$

Ellipsoid

⇒ An ellipsoidal surface can be described as an extension of a spherical surface, where the radii in 3-mutually perpendicular directions can have different values.



The cartesial representation for point over the surface of an ellipsoid centered on the origin is →

$$\left(\frac{x}{r_x}\right)^2 + \left(\frac{y}{r_y}\right)^2 + \left(\frac{z}{r_z}\right)^2 = 1 \quad \text{--- (3)}$$

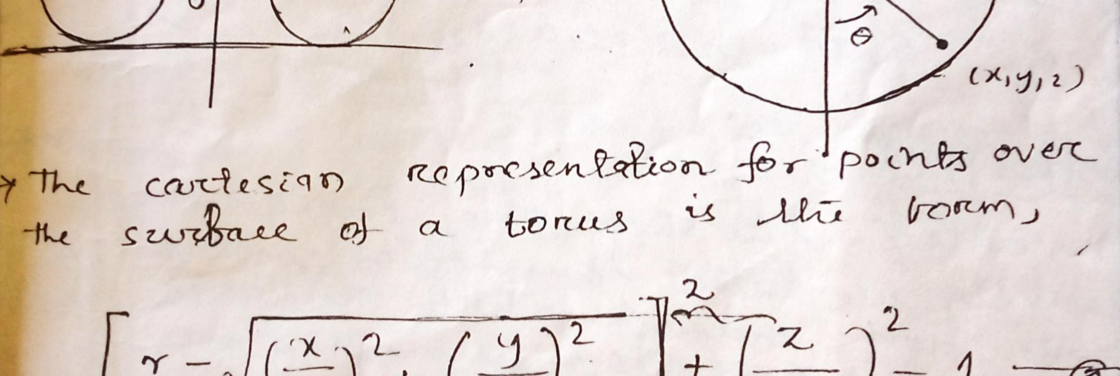
A parametric representation of the ellipsoid in terms of latitude angle ϕ and the longitude angle θ is

$$x = r_x \cos\phi \cos\theta \qquad -\pi/2 \leq \phi \leq \pi/2$$
$$y = r_y \cos\phi \sin\theta \qquad -\pi \leq \theta \leq \pi \quad \text{--- (4)}$$
$$z = r_z \sin\phi$$

Torus

→ A Torus is doughnut-shaped object

→ It can be generated by rotating a circle or other conic about a specified axis.



→ The cartesian representation for points over the surface of a torus is the form,

$$\left[r - \sqrt{\left(\frac{x}{r_x}\right)^2 + \left(\frac{y}{r_y}\right)^2}\,\right]^2 + \left(\frac{z}{r_z}\right)^2 = 1 \quad \text{--- (5)}$$

where, r is any given offset value.

→ ~~The parameter~~

The parametric representation is,

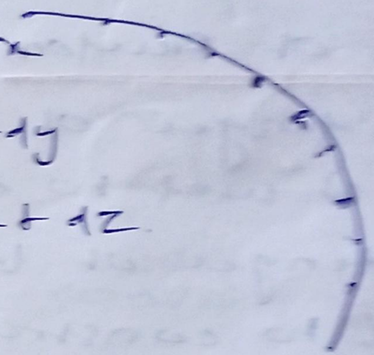
$$x = r_x (r + \cos\phi) \cdot \cos\theta, \quad -\pi \leq \phi \leq \pi$$
$$y = r_y (r + \cos\phi) \cdot \sin\theta, \quad -\pi \leq \theta \leq \pi$$
$$z = r_z \sin\phi$$

— (8)

SP-lin

Sp-line curve:

- Spline is a flexible strip used to produce smooth curve through a designated set of point.
- The curve is described as a piecewise cubic polynomial function whose first and second derivative are continuous across various curve section.
- The spline are used in graphics app for designing of curve and surface shape.



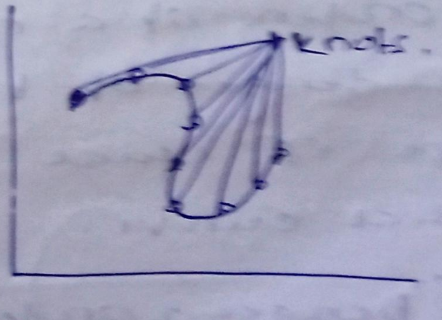
$x(u) = axu^3 + bxu^2 + cxu + dx$

$y(u) = ayu^3 + byu^2 + cyu + dy$

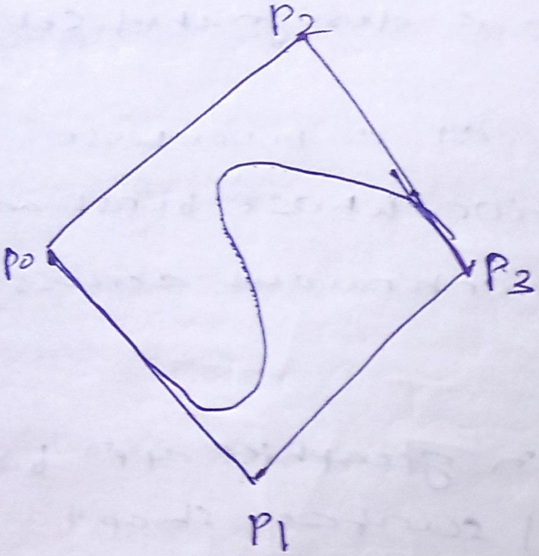
$z(u) = azu^3 + bzu^2 + czu + dz$

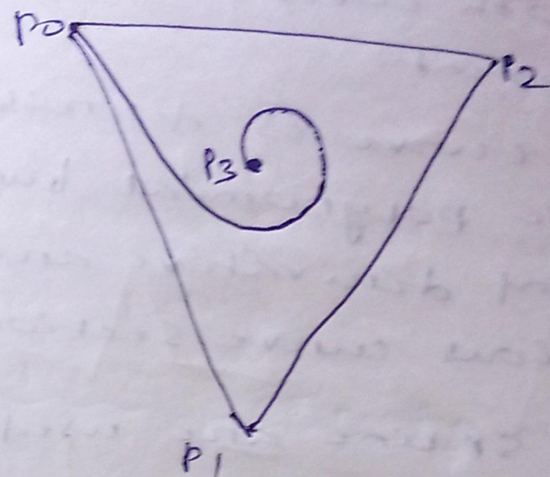
Control point:

To control the shape of a curve interactively we take the help of control points and through which the curve must pass and this is also called as knots.



Convex Hull: It is a polygon boundary that encloses a set of control points.





Degree of freedom:

The degree of freedom is the information which is used to describe the curve.

- The degree of freedom for straight line is 2 (to construct a straight line we require 2 point)
- A curve with more degree of freedom is more complex.

Order of a curve:

Order of a curve deals with the no. of degree of freedom i.e order of a curve is equal to the degree of freedom of that curve.

| Degree = order - 1 |

Order	Degree	Name
1	0	constant
2	1	Any linear eqn
3	2	Qudratic eqn
4	3	cubic surface
5	4	quartic Surface

- For order 2 curve the convex hull is actually a straight line

- For order 3 curve the convex hull is a triangle

- For order 4 curve the convex hull is a quandrilateral.

order of continuity:

Any complex shape is usually not modelled or designed by a straight curve. But we need a several piece of curve for modelling. these joints create a sharpe corner, that can create difficulties in representing curve.

- There are various continuity condn at the joints ① parametric continuity.
② Geometric continuity.

Parametric continuity:

Each piece of curve can be represented by the eqn.

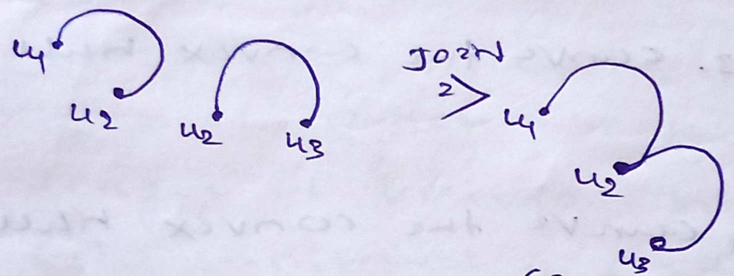
$x = x(u)$ $z = z(u)$

$y = y(u)$ $c = c(u)$ $0 \le u \le 1$

There are 3 types of parametric continuity

Zero order parametric continuity:

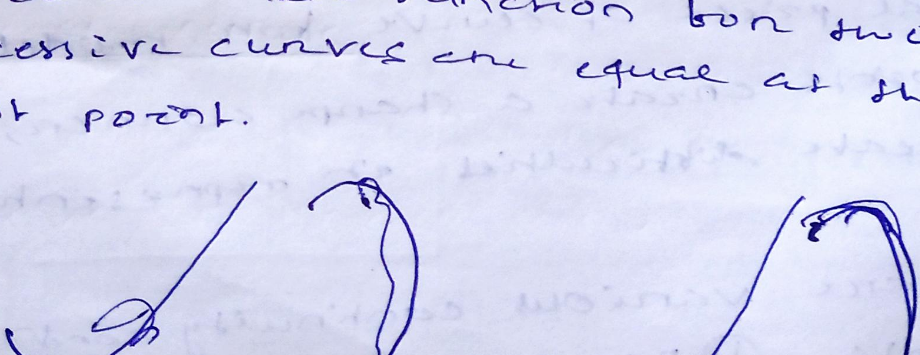
It is defined as c^0. It means that two curve meet with each other

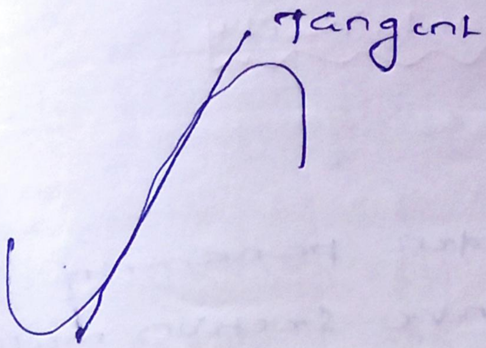


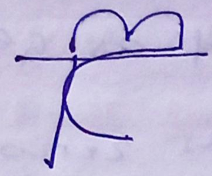
(Resultant u3)

1st order parametric continuity:

It is referred as c^1. It means that 1st parametric derivative (tangent line) of the co-ordinate function for two successive curves are equal at their joint point.



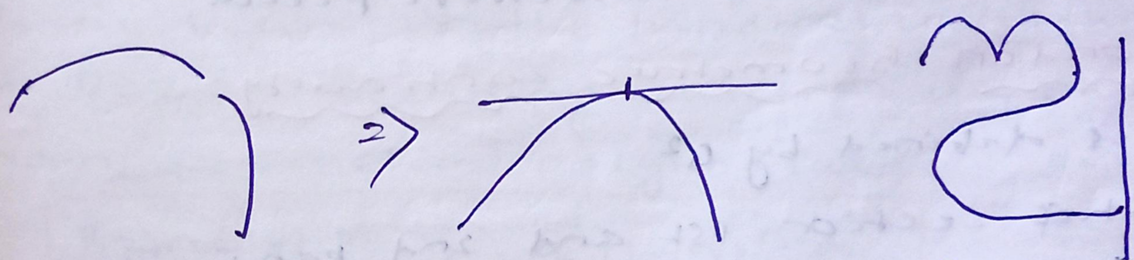




(1st order)

2nd order parametric continuity:

- It is defined by C^2.
- -> Here both the 1st and 2nd parametric derivative of 2 curve section are the same at the intersection.



(2nd order)

Here rate of change of tangent vector for the connecting section are equal at their intersection.

Geometric continuity:

This is an alternative method to join two successive piece of curve.

- Here the parametric derivative of both the pieces should be proportional to each other instead of equal.

Zero order Geometric continuity:

→ It is defined as G^0.

→ It is same as zero order parametric continuity. i.e two curve section must have the same co-ordinate position at the boundary point.

1st order Geometric continuity

- It is denoted by G^1
- Here the 1st order parametric derivative must be proportional to each other at intersection of 2 successive pieces

2nd order Geometric continuity:

- It is defined by G^2
- In this section 1st and 2nd parametric derivative for 2 successive pieces of curves are proportional at joining point and the curvature will match at the intersection point of both the curve pieces.

Bezier curve:

→ Sometimes it is called as Bezier Benstein curve or Bezier curve of cubic polynomial

Defn: Bezier curve is a polynomial of degree 1, less than the no. of control points.

For a given set of (n+1) control points i.e P0, P1, P2 ... Pn the corresponding bezier curve is represented by the equation

$$C(u) = \sum_{i=0}^{n} P_i \, B_{i,n}(u)$$

this equation is known as Bezier polynomial function

where $B_{i,n}$ - is the parameter is known as Bezier blending function or this is known as Bernstein polynomial.

$$B_{i,n}(u) = {}^{n}C_{i} \, u^{i}(1-u)^{n-i}$$

where n = degree.

${}^{n}C_{i}$ -> binomial co-efficient

$${}^{n}C_{i} = \frac{n!}{i!(n-i)!}$$

$$x(u) = \sum_{i=0}^{n} x_i \, B_{i,n}(u)$$

$$y(u) = \sum_{i=0}^{n} y_i \, B_{i,n}(u)$$

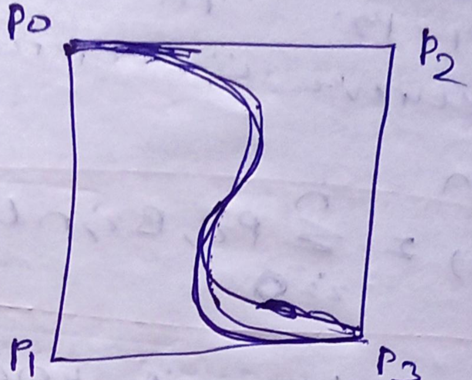
$$z(u) = \sum_{i=0} z_i \, B_{i,n}(u)$$

Properties:

① The bezier curve always passes through the 1st and last point which lies within the convex hull of control points.

1st point: P_0

Last point: P_3



② The curve is tangent to the end points.

③ The curve can be translated, rotated by performing operation on the control points.

④ The sum of Bezier blending function if the curve lies within the convex hull is always 1.

$$\sum_{i=0}^{n} B_{i,n}(u) = 1$$

Q: The Bezier blending function is given as $B_{i,n}(u) = {}^nC_i\, u^i (1-u)^{n-i}$. Generate expression for the blending function of a cubic bezier curve.

Soln:

$${}^nC_i = \frac{n!}{i!(n-i)!}$$

Here $n=3$

So the no. of control points = 4 i.e P_0, P_1, P_2, P_3.

$i = 0$ to 3

$B_{0,3}(u) = {}^3C_0 u^0 (1-u)^{3-0}$

$\Rightarrow$ $\boxed{B_{0,3}(u) = {}^3C_0 (1-u)^3}$

$B_{1,3}(u) = {}^3C_1 u^1 (1-u)^{3-1}$

$\Rightarrow$ $\boxed{B_{1,3}(u) = 3u(1-u)^2}$

$B_{2,3}(u) = {}^3C_2 u^2 (1-u)^{3-2}$

$\Rightarrow B_{2,3}(u) = \frac{3!}{2!\,1!} u^2(1-u)$

$\Rightarrow B_{2,3}(u) = \frac{3 \times 2 \times 1}{2 \times 1 \times 1} u^2(1-u)$

$\Rightarrow$ $\boxed{B_{2,3}(u) = 3u^2(1-u)}$

$B_{3,3}(u) = \frac{3!}{3!\,0!} u^3 (1-u)^{3-3}$

$\Rightarrow$ $\boxed{B_{3,3}(u) = u^3}$

Q: construct enough point on the Bezier curve whose control points are $P_0(4,2)$ $P_1(8,8)$ and $P_2(16,4)$

i) what is the degree of the curve.

ii) what are the co-ordinate at $u=0.5$

degree $n=2$

Soln: degree $n=2$

no. of control point $=3$

Blending function are $B_{0,2}(u)$, $B_{1,2}(u)$, $B_{2,2}(u)$

$B_{0,2}(u) = {}^2C_0 u^0(1-u)^{2-0} = (1-u)^2$

$B_{1,2}(u) = {}^2C_1 u^1(1-u)^{2-1} = 2u(1-u)$

$B_{2,2}(u) = {}^2C_2 u^2(1-u)^{2-2} = u^2$

The bezier curve for the given control points.

$$C(u) = \sum_{i=0}^{2} P_i B_{i,n}(u)$$

$\Rightarrow C(u) = P_0 B_{0,2}(u) + P_1 B_{1,2}(u) + P_2 B_{2,2}(u)$

Desolving this curve in x and y co-ordinate.

$X(u) = X_0 B_{0,2}(u) + X_1 B_{1,2}(u) + X_2 B_{2,2}(u)$

$Y(u) = Y_0 B_{0,2}(u) + Y_1 B_{1,2}(u) + Y_2 B_{2,2}(u)$

Here $X_0 = 4$, $Y_0 = 2$ {as $P_0(4,2)$}

$x_1 = 8$, $y_1 = 8$ } as $P_1(8,8)$

$x_2 = 16$, $y_2 = 4$ } as $P_2(16,4)$

on substitution

$x(u) = 4B_{0,2}(u) + 8B_{1,2}(u) + 16B_{2,2}(u)$

$y(u) = 2B_{0,2}(u) + 8B_{1,2}(u) + 4B_{2,2}(u)$

Now $x(u) = 4(1-u)^2 + 8 \cdot 2u(1-u) + 16u^2$

$y(u) = 2(1-u)^2 + 16(1-u) + 4u^2$

$u = 0.5$

$x(0.5) = 4(1-0.5)^2 + 16 \cdot 0.5(1-0.5) + 16(0.5)^2$

$= 4 \cdot (0.5)^2 + 16 \times \frac{1}{2} \times \frac{1}{2} + 16 \times \frac{1}{2} \times \frac{1}{2}$

$= 4 \times \frac{1}{2} \times \frac{1}{2} + 16 \times \frac{1}{2} \times \frac{1}{2} + 16 \times \frac{1}{2} \times \frac{1}{2}$

$= 1 + 4 + 4 = 9$

$y(0.5) = 2(0.5)^2 + 16 \times 0.5 + 4(0.5)^2$

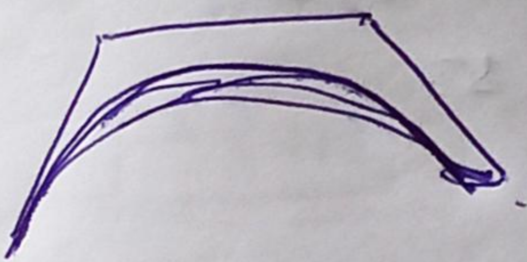
$= 2 \times \frac{1}{2} \times \frac{1}{2} + 16 \times \frac{1}{2} + 4 \times \frac{1}{2} \times \frac{1}{2}$

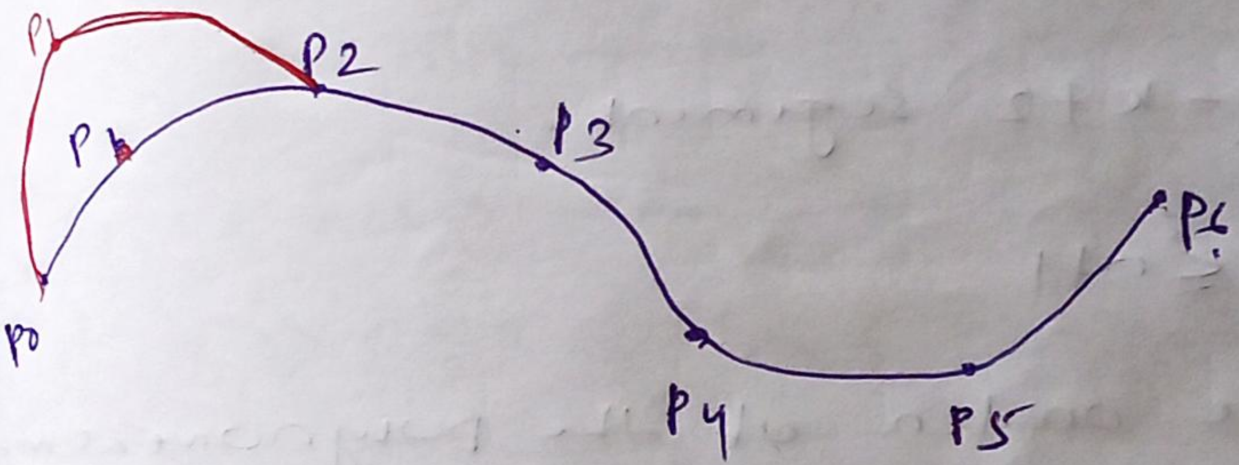
$= 0.5 + 8 + 1$

$= 9.5$

B-spline curve

→





control point = 7.

→ It is a approximation spline curve

→ It has local control over the curve

→ $k = 2 \leq k \leq c.p.$

suppose $k = 3$

C.P = P0 P1 P2 = segement(Q1)

segement	C.P
Q_1	$P_0 P_1 P_2$
Q_2	$P_1 P_2 P_3$
Q_3	$P_2 P_3 P_4$
Q_4	$P_3 P_4 P_5$
Q_5	$P_4 P_5 P_6$

Total segement = $n-k+2$

$= 6-3+2$

$= 5$

$k = 3$

degree = 2

→ It has $n-k+2$ segement.

→ $k = 2 \leq k \leq n+1$

→ k is the order of the polynomial mean of the B-spline curve. order k means that curve k made up piece wise polynomial segement of degree $k-1$.

$$Q(u) = \sum_{i=0}^{n} P_i N_{i,k}(u)$$

P_i = control point (P_0, P_1, P_2)

u = parameter

$$0 \le u \le n-k+2$$

$$x(u) = \sum_{i=0}^{n} x_i N_{i,k}(u)$$

$$y(u) = \sum_{i=0}^{n} y_i N_{i,k}(u)$$

$N_{i,k}(u)$ = B spline function

$$N_{i,k}(u) = \frac{(u-t_i)\,N_{i,k-1}(u)}{t_{i+k-1}-t_i} + \frac{(t_{i+k}-u)\,N_{i+1,k-1}(u)}{t_{i+k}-t_{i+1}}$$

t_i - joint point at which piece of curve join.

t_i is also called knot value

$i \Rightarrow 0 \le i \le n+k \quad (t_0, t_1, t_2 \dots)$

$t_i = 0$ if $i<k$

$t_i = i-k+1$ if $k < i < n$

$t_i = n-k+2$ if $i>n$.

$$N_{i,k}(u) = \begin{cases} 1 & \text{if } t_i \le u \le t_{i+1} \\ 0 & \text{otherwise.} \end{cases}$$

Transformation:

Transformation is the process by which we can change the shape size position dirn of an object w.r.t to a co-ordinate position.

3D Translation:

In 3-D homogeneous representation a point is translated from position $p(x, y, z)$ to $p'(x', y', z')$ with the matrix operation

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

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

So the (t_x, t_y, t_z) are translation vector

$x' = x + t_x$

$y' = y + t_y$

$z' = z + t_z$

So to obtain the inverse translation matrix we simple add a -ve sign to original translation matrix.

Rotation: To perform the rotation we designate an axis of rotation and the amount of angular rotation(θ).

- The positive rotation is always taken as counter clockwise rotation direction about a co-ordinate axis.

Ex: A positive rotation on x-y plane counter clockwise rotation about an axis parallel to z-axis so z-axis the rotational axis.

Rotation about z-axis.

Rotation matrix about z-axis.

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

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

$z' = z$

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

$P' = R_z(\theta) \cdot P$

Rotation about x-axis:

$x' = x$

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

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

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

$\Rightarrow$ $\boxed{P' = R_x(\theta) \cdot P}$

Rotation about y-axis:

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

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

$y' = y$

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

$\boxed{P' = R_y(\theta) \cdot P.}$

Projection: The art of representing a 3-D object on a scene in a 2-D space is known as projection.

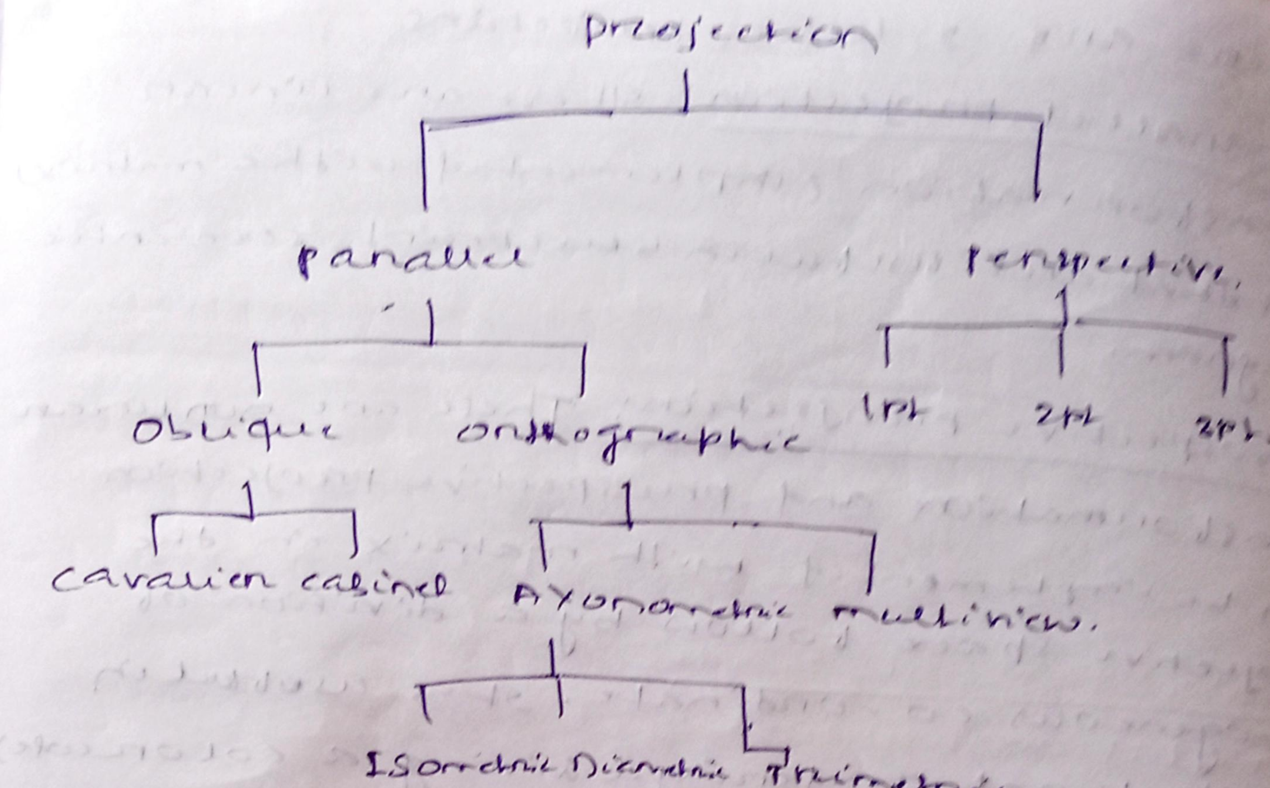
There are 2 basic projections.

① **Parallel projection:** These are linear transformation (implemented with a matrix) and that are useful in blueprint, schematic diagram.

② **Prospective projection:** These are non-linear transformation and prospective projection can be implemented with matrix in the projective space follow by a division of homogeneous co-ordinate. It is useful in architectural rendering. (Giving a color code).

Parallel vs Perspective:

Parallel	Perspective
- less realistic looking	- more realistic
- center of projection is at infinite	- center of projection is at finite.
- projector are parallel to each other	- projector intersect at center of projection
- object size does not change	- size varies inversely with distance
- useful appn requiring the relative properties of an object to be maintained	- Distance and angle are preserved.



parallel projection:

A parallel projection is formed by extending parallel line from each vertex of the object till they intersect the plane of the screen.

Oblique projection:

when the angle between the projector and plane of projection is not equal to 90° then the projection is called oblique projection. Thus projection are not parallel to view plane.

cavalier projection: It is obtained when the angle between the oblique projector and plane of projection is 45° and all three principal axes are equal.

Cabinet projection: Here the line ⊥n to the viewing surface are projected at half of their length.

Orthographic parallel projection: In orthographic projection the projector are ⊥n to the projection plane. or when the angle betn the projection and projector and plane of projection is equal to 90°

Multiview parallel projection:

Here the dirn of projection is ∥el to any of the principal axis.

Axonometric parallel projection:

These are orthographic projection in which the dirn of projection is not parallel to any of the principal axis.

Isometric projection:

The dirn of projection plane make equal angle with each principal axis

Diametric projection:

The projection plane make equal angle with 2 or 3 principal axis.

Trimetric projection:

The projection plane make unequal angle with each principal axis.

Q: A homogeneous co-ordinate point P[3, 2, 1] is translated in x, y and z direction by -2, -2 and -2 unit respectively followed by successive rotation 60° about x-axis. Find the final position of the homogeneous co-ordinate.

Soln: 1) Translation

$$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ t_x & t_y & t_z & 1 \end{bmatrix}$$

$t_x = -2$

$t_y = -2$

$t_z = -2$

$$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -2 & -2 & -2 & 1 \end{bmatrix}$$

2) Rotation 60° about x-axis

$$R = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & \sin\theta & 0 \\ 0 & -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$\theta = 60°$

$$R = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos 60° & \sin 60° & 0 \\ 0 & -\sin 60° & \cos 60° & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0.5 & 0.866 & 0 \\ 0 & -0.866 & 0.5 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Resultant transformation

$T_R = T \times R$

$$= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -2 & -2 & -2 & 1 \end{bmatrix} \times \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0.5 & 0.866 & 0 \\ 0 & -0.866 & 0.5 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_R = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0.5 & 0.866 & 0 \\ 0 & -0.866 & 0.5 & 0 \\ -2 & 0.732 & -2.732 & 1 \end{bmatrix}$$

Final position = Initial position $\times T_R$

$$= P[3,2,1,1] \times \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0.5 & 0.866 & 0 \\ 0 & -0.866 & 0.5 & 0 \\ -2 & 0.732 & -2.732 & 1 \end{bmatrix}$$

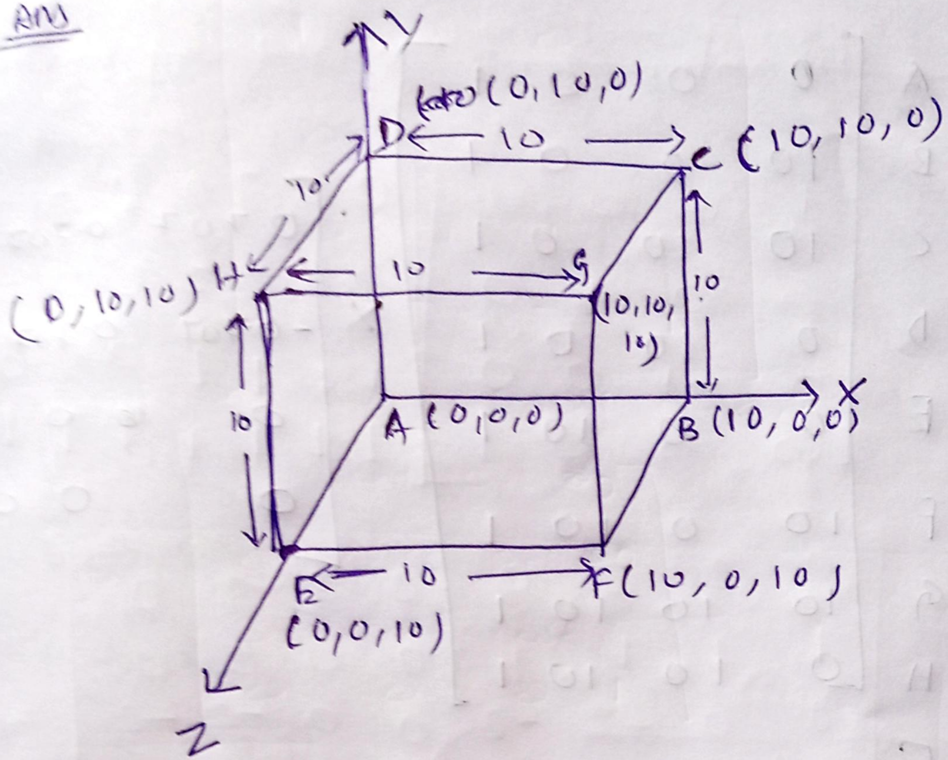
$$= P'[1 \quad 0.864 \quad -0.5 \quad 1]$$

Final position $P'\{1, 0.864, -0.5\}$

Q① A rectangular parallelepiped has its length as 3 unit, 2 unit and 1 unit on x, y and z-axis respectively. perform rotation by 90° clockwise about x-axis.

Q② A cube of length 10 unit having one of its corner at origin (0,0,0) and three edge along 3 principal axis. If the cube is to be rotated about z-axis by angle of 45° in counterclockwise direct, then calculate the new position of the cube

Q② Ans



Soln: Rotation about z-axis by 45° (counter clockwise)

$$R_z = \begin{bmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 0.707 & -0.707 & 0 & 0 \\ 0.707 & 0.707 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Final position of cube

= initial position × R_z

$$
= \begin{matrix} A \\ B \\ C \\ D \\ E \\ F \\ G \\ H \end{matrix}
\begin{bmatrix}
0 & 0 & 0 & 1 \\
10 & 0 & 0 & 1 \\
10 & 10 & 0 & 1 \\
0 & 10 & 0 & 1 \\
0 & 0 & 10 & 1 \\
10 & 0 & 10 & 1 \\
10 & 10 & 10 & 1 \\
0 & 10 & 10 & 1
\end{bmatrix}
\times
\begin{bmatrix}
0.707 & 0.707 & 0 & 0 \\
-0.707 & 0.707 & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1
\end{bmatrix}
$$

$$
= \begin{bmatrix}
0 & 0 & 0 & 1 \\
7.07 & 7.07 & 0 & 1 \\
0 & 14.14 & 0 & 1 \\
-0.707 & 7.07 & 0 & 1 \\
0 & 0 & 10 & 1 \\
7.07 & 7.07 & 10 & 1 \\
0 & 1.414 & 10 & 1 \\
-7.07 & 7.07 & 10 & 1
\end{bmatrix}
$$

Q(1) Ans

A → 0 0 1

B → 3 0 1

C → 3 2 1

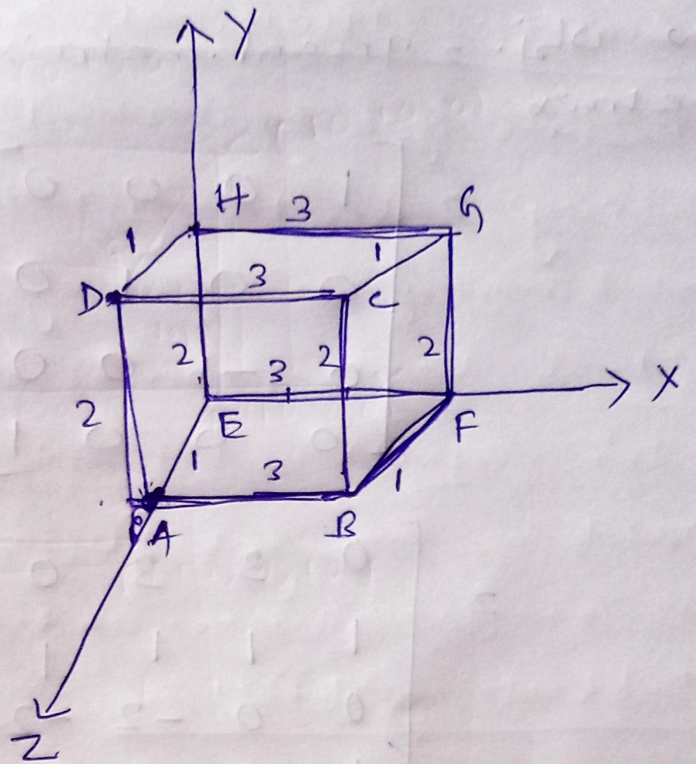
D → 0 2 1

E → 0 0 0

F → 3 0 0

G → 3 2 0

H → 0 2 0.



For rotation about x-axis

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(-90°) & -\sin(-90°) & 0 \\ 0 & \sin(-90°) & \cos(-90°) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Old object matrix =

$$\begin{bmatrix} 0 & 3 & 3 & 0 & 0 & 3 & 3 & 0 \\ 0 & 0 & 2 & 2 & 0 & 0 & 2 & 2 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

new obj. = transt. matrix × old obj matrix
matrix

$$= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 3 & 3 & 0 & 0 & 3 & 3 & 0 \\ 0 & 0 & 2 & 2 & 0 & 0 & 2 & 2 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 3 & 3 & 0 & 0 & 3 & 3 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -2 & -2 & 0 & 0 & -2 & -2 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$