B.C.A study

Unit-1: Introduction to CGMA

Advantages of interactive graphics

In interactive Computer Graphics user have some controls over the picture, i.e., the user can make any change in the produced image.

Interactive Computer Graphics require two-way communication between the computer and the user. A User can see the image and make any change by sending his command with an input device.

Advantages:

1. Higher Quality
2. More precise results or products
3. Greater Productivity
4. Lower analysis and design cost
5. Significantly enhances our ability to understand data and to perceive trends.

Representative use of computer graphics

The use of computer graphics is wide spread. It is used in various areas such as industry, business, government organizations, education, entertainment and most recently the home. User friendliness is one of the main factors underlying the success and popularity of any system. It is now a well established fact that graphical interfaces provide in attractive and easy interaction between users and computers.

In industry, business, government and educational organizations, computer graphics is most commonly used to create 2D and 3D graphics of mathematical, physical and economic functions in form of histograms, bars, and pie-chats. These graphs and charts are very useful for decision making.
The desktop publishing on personal computers allow the use of graphics for the creation and

dissemination of information. Many organizations does the in-house creation and dissemination of documents. The desktop publishing allows user to create documents which contain text, tables, graphs, and other forms of drawn or scanned images or pictures. This is one approach towards the office automation.The computer-aided drafting uses graphics to design components and systems electrical, mechanical, electromechanical and electronic devices such as automobile bodies, structures of building, airplane, slips, very large-scale integrated chips, optical systems and computer networks.
Use of graphics in simulation makes mathematic models and mechanical systems more realistic and easy to study. The interactive graphics supported by animation software proved their use in production of animated movies and cartoons films.
There is lot of development in the tools provided by computer graphics. This allows user to create artistic pictures which express messages and attract attentions. Such pictures are very useful in advertising.

Conceptual Framework for Interactive Graphics

Conceptual Framework has the following elements:

– Graphics Library – Between application and display hardware there is graphics library / API.

– Application Program – An application program maps all application objects to images by invoking graphics.

– Graphics System – An interface that interacts between Graphics library and Hardware.

– Modifications to images are the result of user interaction.

Scan Converting a Straight Line

A straight line may be defined by two endpoints & an equation. In fig the two endpoints are described by (x_1,y_1) and (x_2,y_2). The equation of the line is used to determine the x, y coordinates of all the points that lie between these two endpoints.

Scan Converting a Straight Line

[bcastudyguide.com (https://bcastudyguide.wordpress.com)](https://bcastudyguide.wordpress.com)

Using the equation of a straight line, y = mx + b where m = Scan Converting a Straight Line & b = the y interrupt, we can find values of y by incrementing x from $x = x_1$, to $x = x_2$. By scan-converting these calculated x, y values, we represent the line as a sequence of pixels.

Algorithm for drawing line using equation:

Step1: Start Algorithm

Step2: Declare variables x_1,x_2,y_1,y_2,dx,dy,m,b,

Step3: Enter values of x_1,x_2,y_1,y_2.
The (x_1,y_1) are co-ordinates of a starting point of the line.
The (x_2,y_2) are co-ordinates of a ending point of the line.

Step4: Calculate $dx = x_2 - x_1$

Step5: Calculate $dy = y_2 - y_1$

Step6: Calculate m = Scan Converting a Straight Line

Step7: Calculate $b = y_1 - m * x_1$

Step8: Set (x, y) equal to starting point, i.e., lowest point and x_{end} equal to largest value of x.

If dx < 0
then $x = x_2$
$y = y_2$
$x_{end} = x_1$
If dx > 0
then $x = x_1$
$y = y_1$
$x_{end} = x_2$

Step9: Check whether the complete line has been drawn if $x=x_{end}$, stop

Step10: Plot a point at current (x, y) coordinates

Step11: Increment value of x, i.e., x = x+1

Step12: Compute next value of y from equation y = mx + b

Step13: Go to Step9.

Program to draw a line using LineSlope Method

```
#include <graphics.h>
#include <stdlib.h>
#include <math.h>
#include <stdio.h>
#include <conio.h>
#include <iostream.h>

class bresen
{
   float x, y, x1, y1, x2, y2, dx, dy, m, c, xend;
   public:
   void get ();
   void cal ();
};
   void main ()
   {
   bresen b;
   b.get ();
   b.cal ();
   getch ();
  }
   Void bresen :: get ()
  {
   print (“Enter start & end points”);
   print (“enter x1, y1, x2, y2”);
   scanf (“%f%f%f%f”,sx1, sx2, sx3, sx4)
}
void bresen ::cal ()
{
   /* request auto detection */
   int gdriver = DETECT,gmode, errorcode;
   /* initialize graphics and local variables */
   initgraph (&gdriver, &gmode, ” “);
   /* read result of initialization */
   errorcode = graphresult ();
```

```
if (errorcode ! = grOK)    /*an error occurred */
{
   printf("Graphics error: %s \n", grapherrormsg (errorcode);
   printf ("Press any key to halt:");
   getch ();
   exit (1); /* terminate with an error code */
}
dx = x2-x1;
dy=y2-2y1;
m = dy/dx;
c = y1 – (m * x1);
if (dx<0)
{
   x=x2;
   y=y2;
   xend=x1;
}
else
{
   x=x1;
   y=y1;
   xend=x2;
}
while (x<=xend)
{
   putpixel (x, y, RED);
   y++;
   y=(x*x) +c;
}
}
```

scan converting a Circle:

Circle is an eight-way symmetric figure. The shape of circle is the same in all quadrants. In each quadrant, there are two octants. If the calculation of the point of one octant is done, then the other seven points can be calculated easily by using the concept of eight-way symmetry.

For drawing, circle considers it at the origin. If a point is $P_1(x, y)$, then the other seven points will be

Defining a Circle

So we will calculate only 45°arc. From which the whole circle can be determined easily.

If we want to display circle on screen then the putpixel function is used for eight points as shown below:

putpixel (x, y, color)
putpixel (x, -y, color)
putpixel (-x, y, color)
putpixel (-x, -y, color)
putpixel (y, x, color)
putpixel (y, -x, color)
putpixel (-y, x, color)
putpixel (-y, -x, color)

Example: Let we determine a point (2, 7) of the circle then other points will be (2, -7), (-2, -7), (-2, 7), (7, 2), (-7, 2), (-7, -2), (7, -2)

These seven points are calculated by using the property of reflection. The reflection is accomplished in the following way:

The reflection is accomplished by reversing x, y co-ordinates.

Defining a Circle

There are two standards methods of mathematically defining a circle centered at the origin.

1. Defining a circle using Polynomial Method
2. Defining a circle using Polar Co-ordinates

Program to draw a circle using Polynomial Method:

```
#include<graphics.h>
#include<conio.h>
#include<math.h>
voidsetPixel(int x, int y, int h, int k)
{
    putpixel(x+h, y+k, RED);
    putpixel(x+h, -y+k, RED);
    putpixel(-x+h, -y+k, RED);
    putpixel(-x+h, y+k, RED);
    putpixel(y+h, x+k, RED);
    putpixel(y+h, -x+k, RED);
    putpixel(-y+h, -x+k, RED);
    putpixel(-y+h, x+k, RED);
}
main()
{
    intgd=0, gm,h,k,r;
    double x,y,x2;
    h=200, k=200, r=100;
    initgraph(&gd, &gm, "C:\\TC\\BGI ");
    setbkcolor(WHITE);
    x=0,y=r;
```

```
x2 = r/sqrt(2);
while(x<=x2)
{
   y = sqrt(r*r – x*x);
   setPixel(floor(x), floor(y), h,k);
   x += 1;
}
getch();
closegraph();
return 0;
}
```

Program to draw a circle using Polar Coordinates:

```
#include <graphics.h>
#include <stdlib.h>
#define color 10
void eightWaySymmetricPlot(int xc,int yc,int x,int y)
{
   putpixel(x+xc,y+yc,color);
   putpixel(x+xc,-y+yc,color);
   putpixel(-x+xc,-y+yc,color);
   putpixel(-x+xc,y+yc,color);
   putpixel(y+xc,x+yc,color);
   putpixel(y+xc,-x+yc,color);
   putpixel(-y+xc,-x+yc,color);
   putpixel(-y+xc,x+yc,color);
}
void PolarCircle(int xc,int yc,int r)
{
   int x,y,d;
   x=0;
   y=r;
   d=3-2*r;
   eightWaySymmetricPlot(xc,yc,x,y);
   while(x<=y)
   {
      if(d<=0)
      {
         d=d+4*x+6;
      }
      else
      {
         d=d+4*x-4*y+10;
         y=y-1;
      }
```

```
    x=x+1;
    eightWaySymmetricPlot(xc,yc,x,y);
  }
}
int main(void)
{
  int gdriver = DETECT, gmode, errorcode;
  int xc,yc,r;
  initgraph(&gdriver, &gmode, “c:\\turboc3\\bgi”);
errorcode = graphresult();
if (errorcode != grOk)
  {
    printf(“Graphics error: %s\n”, grapherrormsg(errorcode));
    printf(“Press any key to halt:”);
    getch();
    exit(1);
  }
printf(“Enter the values of xc and yc ,that is center points of circle : “);
  scanf(“%d%d”,&xc,&yc);
  printf(“Enter the radius of circle : “);
  scanf(“%d”,&r);
  PolarCircle(xc,yc,r);
  getch();
  closegraph();
  return 0;
}
```

Scan Converting a Ellipse:

The ellipse is also a symmetric figure like a circle but is four-way symmetry rather than eight-way.

Scan Converting a Ellipse

Program to Implement Ellipse Drawing Algorithm:

```
#include<stdio.h>
#include<conio.h>
#include<graphics.h>
#include<math.h>
void disp();
float x,y;
intxc,yc;
void main()
{
        intgd=DETECT,gm,a,b;
        float p1,p2;
        clrscr();
        initgraph(&gd,&gm,"c:\\turboc3\\bgi");
        printf("*** Ellipse Generating Algorithm ***\n");
        printf("Enter the value of Xc\t");
        scanf("%d",&xc);
        printf("Enter the value of yc\t");
        scanf("%d",&yc);
        printf("Enter X axis length\t");
        scanf("%d",&a);
        printf("Enter Y axis length\t");
        scanf("%d",&b);
        x=0;y=b;
        disp();
        p1=(b*b)-(a*a*b)+(a*a)/4;
        while((2.0*b*b*x)<=(2.0*a*a*y))
        {
                x++;
                if(p1<=0)
                p1=p1+(2.0*b*b*x)+(b*b);
                else
    {
                        y–;
                        p1=p1+(2.0*b*b*x)+(b*b)-(2.0*a*a*y);
    }
                disp();
                x=-x;
                disp();
                x=-x;
                delay(50);
        }
        x=a;
        y=0;
        disp();
        p2=(a*a)+2.0*(b*b*a)+(b*b)/4;
```

```
            while((2.0*b*b*x)>(2.0*a*a*y))
            {
                    y++;
                    if(p2>0)
                    p2=p2+(a*a)-(2.0*a*a*y);
                    else
        {
                            x–;
                            p2=p2+(2.0*b*b*x)-(2.0*a*a*y)+(a*a);
        }
                    disp();
                    y=-y;
                    disp();
                    y=-y;
                    delay(50);
          }
            getch();
            closegraph();
}
void disp()
{
           putpixel(xc+x,yc+y,7);
            putpixel(xc-x,yc+y,7);
            putpixel(xc+x,yc-y,7);
       putpixel(xc+x,yc-y,7);
  }
```

B.C.A study

Unit-2: Useful technologies related to computer graphics

HARDCOPIES TECHNOLOGIES

Hardcopy is a printed copy of information from a computer. Sometimes it refer to as a printout, so it called hardcopy because it exists as a physical object. Hardcopy is tangible output that usually printed. The principal examples are printouts, whether text or graphics, form printers and also films including microfilms and microfirche is also considered as hardcopy output.

Output Devices

It is an electromechanical device, which accepts data from a computer and translates them into form understand by users.

Following are Output Devices:

1. Printers
2. Plotters

Printers:

Printer is the most important output device, which is used to print data on paper.

1. Impact Printers: The printers that print the characters by striking against the ribbon and onto the papers are known as Impact Printers.

These Printers are of two types:

1. Character Printers
2. Line Printers

2. Non-Impact Printers: The printers that print the characters without striking against the ribbon and onto the papers are called Non-Impact Printers. These printers print a complete page at a time, therefore, also known as Page Printers.

Page Printers are of two types:

1. Laser Printers
2. Inkjet Printers

LASER PRINTER : laser printer is a type of printer which makes use of a laser beam to produce an image on the drum or you can say that it is a printer that makes use of a focused beam of light to transfer text and images onto paper. Laser printer makes use of laser technology for the printing of copies onto the paper.

INKJET PRINTER : Inkjet printers are a category of printer in which printing is done with the help of inkjet technology. The technology works by spraying ionized ink directed by magnetic plates onto the paper, which is fed through the printer.

DISPLAY TECHNOLOGIES

Visual Displays – Basic Technologies

- Cathode Ray Tubes
- Flat Panel Displays
- Electroluminescence Displays
- LCD Displays

- Active Matrix TFT
- Light Valves
- Micro mirror devices.

Raster-Scan Displays

Raster Scan Displays are most common type of graphics monitor which employs CRT. It is based on television technology. In raster scan system electron beam sweeps across the screen, from top to bottom covering one row at a time.A pattern of illuminated pattern of spots is created by turning beam intensity on and off as it moves across each row. A memory area called refresh buffer or frame buffer stores picture definition. This memory area holds intensity values for all screen points. Stored intensity values are restored from frame buffer and painted on screen taking one row at a time.Each screen point is referred to as pixels.

In raster scan systems refreshing is done at done at a rate of 60-80 frames per second. Refresh rates are also sometimes described in units of cycles per second / Hertz (Hz). At the end of each scan line, electron beam begins to display next scan line after returning to left side of screen. The return to the left of screen after refresh of each scan line is known as *horizontal retrace* of electron beam. At the end of each frame electron beam returns to top left corner and begins the next frame.

Raster-Scan Display Processor:
An important function of display process is to digitize a picture definition given in an application program into a set of pixel-intensity values for storage in refresh buffer. This process is referred to as **scan conversion**. The purpose of display processors is to relieve the CPU from graphics jobs.

Display processors can perform various other tasks like: creating different line styles, displaying color areas, etc. Typically display processors are utilized to interface input devices, such as mouse, joysticks.

Random-Scan Display

In **Random-Scan Display** electron beam is directed only to the ares of screen where a picture has to be drawn. It is also called vector displays, as it draws picture one line at time. It can draw and refresh component lines of a picture in any specified sequence. Pen plotter is an example of random-scan displays.

The number of lines regulates refresh rate on random-scan displays. An area of memory called **refresh display files** stores picture definition as a set of line drawing commands. The system returns back to first line command in the list, after all the drawing commands have been processed. High-quality vector systems can handle around 100, 00 short lines at this refresh rate. Faster refreshing can burn the phosphor. To avoid this every refresh cycle is delayed to prevent refresh rate greater than 60 frames per second.

Random-Scan Display Processors:

Input in the form of an application program is stored in the system memory along with graphics package. Graphics package translates the graphic commands in application program into a display file stored in system memory. This display file is then accessed by the display processor to refresh the screen. The display processor cycles through each command in the display file program. Sometimes the display processor in a random-scan is referred as *Display Processing Unit* / *Graphics Controller*.

VIDEO CONTROLLER:

A video controller, often referred to as a video or graphics card, is a key hardware component that allows computers to generate graphic information to any video display devices, such as a monitor or projector. They are also known as graphics or video adapters. Some modern computers do not include video cards, but rather have graphics processing units directly integrated into the computer's motherboard.

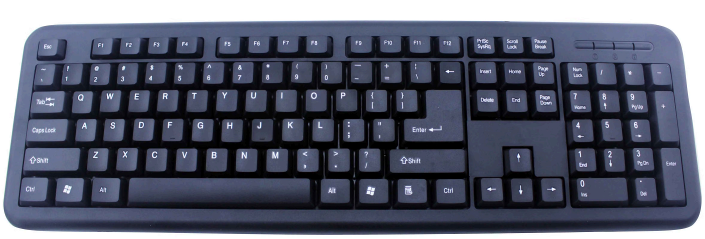
Following are some of the important input devices which are used in a computer −

- Keyboard
- Mouse
- Joy Stick
- Light pen
- Track Ball
- Scanner

Keyboard

Keyboard is the most common and very popular input device which helps to input data to the computer. The layout of the keyboard is like that of traditional typewriter, although there are some additional keys provided for performing additional functions.

Keyboards are of two sizes 84 keys or 101/102 keys, but now keyboards with 104 keys or 108 keys are also available for Windows and Internet.



bcastudyguide.com (https://bcastudyguide.wordpress.com)

Mouse

Mouse is the most popular pointing device. It is a very famous cursor-control device having a small palm size box with a round ball at its base, which senses the movement of the mouse and sends corresponding signals to the CPU when the mouse buttons are pressed.

bcastudyguide.com (https://bcastudyguide.wordpress.com)

Generally, it has two buttons called the left and the right button and a wheel is present between the buttons. A mouse can be used to control the position of the cursor on the screen, but it cannot be used to enter text into the computer.

Joystick

Joystick is also a pointing device, which is used to move the cursor position on a monitor screen. It is a stick having a spherical ball at its both lower and upper ends. The lower spherical ball moves in a socket. The joystick can be moved in all four directions.

bcastudyguide.com (https://bcastudyguide.wordpress.com)

The function of the joystick is similar to that of a mouse. It is mainly used in Computer Aided Designing (CAD) and playing computer games.

Light Pen

Light pen is a pointing device similar to a pen. It is used to select a displayed menu item or draw pictures on the monitor screen. It consists of a photocell and an optical system placed in a small tube.

bcastudyguide.com (https://bcastudyguide.wordpress.com)

When the tip of a light pen is moved over the monitor screen and the pen button is pressed, its photocell sensing element detects the screen location and sends the corresponding signal to the CPU.

Track Ball

Track ball is an input device that is mostly used in notebook or laptop computer, instead of a mouse. This is a ball which is half inserted and by moving fingers on the ball, the pointer can be moved.

[bcastudyguide.com (https://bcastudyguide.wordpress.com)]

Since the whole device is not moved, a track ball requires less space than a mouse. A track ball comes in various shapes like a ball, a button, or a square.

Scanner

Scanner is an input device, which works more like a photocopy machine. It is used when some information is available on paper and it is to be transferred to the hard disk of the computer for further manipulation.

[bcastudyguide.com (https://bcastudyguide.wordpress.com)]

Scanner captures images from the source which are then converted into a digital form that can be stored on the disk. These images can be edited before they are printed.

WORKING EXPOSURE

‘Working exposure’ means ‘working for free’ in exchange for exposure or experience.

Dreamweaver is Adobe’s flagship tool, which primarily focuses on creating and editing HTML and CSS, while giving a preview of the output. Web designing and application deveopment can be done simultaneously, with real-time checking and editing.

You, possibly, must have seen this phrase in a Multimedia course curriculum where the university/college/teaching body will give you a hands-on experience with tools such as the Dreamweaver.

CLIPPING

Clipping, in the context of computer graphics, is a method to selectively enable or disable rendering operations within a defined region of interest. Mathematically, clipping can be described using the terminology of constructive geometry. A rendering algorithm only draws pixels in the intersection between the clip region and the scene model. Lines and surfaces outside the view volume (aka. frustum) are removed.

Clip regions are commonly specified to improve render performance. A well-chosen clip allows the renderer to save time and energy by skipping calculations related to pixels that the user cannot see. Pixels that will be drawn are said to be within the clip region. Pixels that will not be drawn are outside the clip region. More informally, pixels that will not be drawn are said to be “clipped.”

Clipping in 3D graphics:

In three-dimensional graphics, the terminology of clipping can be used to describe many related features. Typically, “clipping” refers to operations in the plane that work with rectangular shapes, and “culling” refers to more general methods to selectively process scene model elements. This terminology is not rigid, and exact usage varies among many sources.

Scene model elements include geometric primitives: points or vertices; line segments or edges; polygons or faces; and more abstract model objects such as curves, splines, surfaces, and even text. In complicated scene models, individual elements may be selectively disabled (clipped) for reasons including visibility within the viewport (frustum culling); orientation (backface culling), obscuration by other scene or model elements (occlusion culling, depth- or “z” clipping). Sophisticated algorithms exist to efficiently detect and perform such clipping. Many optimized clipping methods rely on specific hardware acceleration logic provided by a graphics processing unit (GPU).

The concept of clipping can be extended to higher dimensionality using methods of abstract algebraic geometry

Cohen–Sutherland algorithm

The **Cohen–Sutherland algorithm** is a computer-graphics algorithm used for line clipping. The algorithm divides a two-dimensional space into 9 regions and then efficiently determines the lines and portions of lines that are visible in the central region of interest (the viewport).

The algorithm was developed in 1967 during flight-simulator work by Danny Cohen and Ivan Sutherland.

The algorithm includes, excludes or partially includes the line based on whether:

- Both endpoints are in the viewport region (bitwise OR of endpoints = 00): trivial accept.
- Both endpoints share at least one non-visible region, which implies that the line does not cross the visible region. (bitwise AND of endpoints ≠ 0): trivial reject.
- Both endpoints are in different regions: in case of this nontrivial situation the algorithm finds one of the two points that is outside the viewport region (there will be at least one point outside). The intersection of the outpoint and extended viewport border is then calculated (i.e. with the parametric equation for the line), and this new point replaces the outpoint. The algorithm repeats until a trivial accept or reject occurs.

figure below are called outcodes. An outcode is computed for each of the two points in the line. The outcode will have 4 bits for two-dimensional clipping, or 6 bits in the three-dimensional case. The first bit is set to 1 if the point is above the viewport. The bits in the 2D outcode represent: top, bottom, right, left. For example, the outcode 1010 represents a point that is top-right of the viewport.

	left	**central**	**right**
top	1001	1000	1010
central	0001	0000	0010
bottom	0101	0100	0110

Cyrus–Beck algorithm

The **Cyrus–Beck algorithm** is a generalized line clipping algorithm. It was designed to be more efficient than the Cohen–Sutherland algorithm, which uses repetitive clipping.[1] **(https://en.wikipedia.org/wiki/Cyrus%E2%80%93Beck_algorithm#cite_note-1)** Cyrus–Beck is a general algorithm and can be used with a convex polygon clipping window, unlike Sutherland–Cohen, which can be used only on a rectangular clipping area.

Here the parametric equation of a line in the view plane is

0\le
t\lec
where {\displaystyle 0\leq t\leq 1}1 .

Now to find the intersection point with the clipping window, we calculate the value of the dot product. Let $\mathbf{p}_E$ be a point on the clipping plane E.

{\dis
\ma
{n}
\cdc
(\ma
{p}
(t)-
\ma
{p}
Calculate {\displaystyle \mathbf {n} \cdot (\mathbf {p} (t)-\mathbf {p} _{E})}_{E}):

if < 0, vector pointed towards interior;

if = 0, vector pointed parallel to plane containing p;

if > 0, vector pointed away from interior.

Here **n** stands for normal of the current clipping plane (pointed away from interior).

By this we select the point of intersection of line and clipping window where (dot product is 0) and hence clip the line.

Midpoint Subdivision Algorithm:-

Midpoint subdivision algorithm is an extension of the Cyrus Beck algorithm. This algorithm is mainly used to compute visible areas of lines that are present in the view port are of the sector or the image. It follows the principle of the bisection method and works similarly to the Cyrus Beck algorithm by bisecting the line in to equal halves. But unlike the Cyrus Beck algorithm, which only bisects the line once, Midpoint Subdivision Algorithm bisects the line numerous times.

Also the Sutherland Cohen subdivision line clipping algorithm requires the calculation of the intersection of the line with the window edge. These calculations can be avoided by repetitively subdividing the line at its midpoint.

Step1: Calculate the position of both endpoints of the line

Step2: Perform OR operation on both of these endpoints

Step3: If the OR operation gives 0000
then
Line is guaranteed to be visible
else
Perform AND operation on both endpoints.
If AND ≠ 0000
then the line is invisible
else
AND=6000
then the line is clipped case.

Step4: For the line to be clipped. Find midpoint
$X_m=(x_1+x_2)/2$
$Y_m=(y_1+y_2)/2$
X_m is midpoint of X coordinate.
Y_m is midpoint of Y coordinate.

Step5: Check each midpoint, whether it nearest to the boundary of a window or not.

Step6: If the line is totally visible or totally rejected not found then repeat step 1 to 5.

Step7: Stop algorithm.

•••

B.C.A study

Unit-3: Geometric transformation

A geometric transformation in computer graphics and multimedia applications refers to a change in position, size, shape or orientation of an object or image. Some common examples of geometric transformations include translation, rotation, scaling and reflection. These transformations can be performed using mathematical algorithms and matrices to manipulate the coordinates of an object in a 2D or 3D space. The result of these transformations can be visualized on a computer screen to produce the desired effect.

2D Transformation

2D transformations refer to the transformations applied to objects or images in a 2D plane. Some common types of 2D transformations include:

1. Translation: A translation moves an object from one position to another without changing its orientation or size.
2. Rotation: A rotation changes the orientation of an object around a fixed point, known as the origin.
3. Scaling: A scaling transformation changes the size of an object, either uniformly (isotropic scaling) or non-uniformly (anisotropic scaling).
4. Shear: A shear transformation changes the shape of an object, moving its parallel sides closer together or further apart.
5. Reflection: A reflection transformation creates a mirror image of an object across a specified axis.

These transformations are typically performed using matrix operations, where the object's coordinates are transformed according to a transformation matrix. The transformed coordinates are then used to render the object in its new position, orientation, size, or shape

Homogeneous Coordinates and Matrix Representation of 2D
Transformations

Homogeneous coordinates are a method of representing 2D points and vectors in a three-dimensional space. In homogeneous coordinates, each 2D point (x, y) is represented as a 3D point (x, y, w) where w ≠ 0. This allows for more concise and efficient representation of certain types of transformations, such as translations and projective transformations.

The matrix representation of 2D transformations using homogeneous coordinates is represented by 3×3 matrices, where the last row is always [0, 0, 1]. The matrix multiplication between the transformation matrix and the homogeneous coordinates of a point results in the transformed coordinates of the point.

For example, the translation matrix for a translation by (dx, dy) can be represented as:

[1 0 dx]
[0 1 dy]
[0 0 1]

The rotation matrix for a rotation by angle θ can be represented as:

[cos(θ) -sin(θ) 0]
[sin(θ) cos(θ) 0]
[0 0 1]

The scaling matrix for a uniform scaling by factor s can be represented as:

[s 0 0]
[0 s 0]
[0 0 1]

By using matrix representation of transformations, multiple transformations can be combined into a single matrix, allowing for efficient and concise representation of complex transformations.

representation of 2d transformation

The matrix representation of 2D transformations is a mathematical way of representing the changes that occur to an object as a result of a transformation. The transformation matrix, which is a 2×2 or 3×3 matrix, defines how the coordinates of an object are transformed. The transformation is then performed by matrix multiplication between the transformation matrix and the original coordinates of the object.

For example, the matrix representation of a 2D translation can be given as:

[1 0 dx]
[0 1 dy]

Where (dx, dy) represents the translation vector. To perform the translation, the original coordinates (x, y) are multiplied with the transformation matrix:

[x′] [1 0 dx] [x]
[y′] = [0 1 dy] * [y]
[1] [0 0 1] [1]

The result (x′, y′) are the transformed coordinates of the object.

Similarly, the matrix representation of a 2D rotation can be given as:

[cos(θ) -sin(θ)]
[sin(θ) cos(θ)]

Where θ is the angle of rotation. To perform the rotation, the original coordinates (x, y) are multiplied with the transformation matrix:

[x′] [cos(θ) -sin(θ)] [x]
[y′] = [sin(θ) cos(θ)] * [y]

The result (x′, y′) are the transformed coordinates of the object.

The matrix representation of a 2D scaling can be given as:

[sx 0]
[0 sy]

Where (sx, sy) represents the scaling factors along the x and y axes, respectively. To perform the scaling, the original coordinates (x, y) are multiplied with the transformation matrix:

[x′] [sx 0] [x]
[y′] = [0 sy] * [y]

The result (x′, y′) are the transformed coordinates of the object.

composition of 2D transformations

The composition of 2D transformations refers to the application of multiple transformations to an object, one after the other, to achieve a desired result. The order in which the transformations are applied can have a significant impact on the final result.

In matrix representation, the composition of transformations can be represented as matrix multiplication, where the matrices representing each transformation are multiplied together in the desired order.

For example, if we have a translation matrix T, a rotation matrix R, and a scaling matrix S, the composition of these transformations can be represented as:

A = S * R * T

This means that first, a translation is applied (T), then a rotation (R), and finally a scaling (S).

To apply the transformations to a given point (x, y), we simply multiply the point with the composition matrix A:

[x'] [S * R * T] [x]
[y'] = [ ] * [y]
[1] [ ] [1]

The result (x', y') are the transformed coordinates of the point, reflecting the combined effect of all three transformations.

It's important to note that the order of transformations affects the final result, so it's important to choose the right order depending on the desired outcome. In some cases, it may be necessary to use the inverse of a transformation matrix to undo its effect, or to switch the order of transformations.

the Window-to-Viewport Transformations

Window-to-Viewport transformations are used in computer graphics to map a 2D world-coordinate system (the "window") to a 2D screen-coordinate system (the "viewport"). This mapping is necessary to display the graphical objects in a way that fits the screen.

The window-to-viewport transformation can be represented as a matrix multiplication of the form:

[x'] [sx 0 tx] [x]
[y'] = [0 sy ty] * [y]
[1] [0 0 1] [1]

Where:

- (x, y) are the original window coordinates of a point
- (x', y') are the transformed viewport coordinates of the point
- (sx, sy) are the scaling factors along the x and y axes, respectively
- (tx, ty) are the translation factors along the x and y axes, respectively

The scaling factors (sx, sy) are used to stretch or shrink the window coordinates so that they fit the viewport. The translation factors (tx, ty) are used to shift the origin of the window coordinates to a different location within the viewport.

It's important to note that the window-to-viewport transformation does not change the aspect ratio of the objects. This means that the relative proportions of the objects are preserved, even though their size may change. This is important to ensure that the objects are displayed correctly on the screen, without being distorted.

Introduction to 3D Transformations Matrix.

3D transformations are used in computer graphics to manipulate 3D objects in a virtual world. A 3D transformation matrix is used to represent a 3D transformation, just like a 2D transformation matrix represents a 2D transformation.

A 3D transformation matrix can represent a combination of translation, rotation, and scaling operations in 3D space. The matrix representation of a 3D transformation allows for efficient computation and application of the transformations to 3D objects.

For example, a 3D translation matrix can be represented as:

[1 0 0 tx]
[0 1 0 ty]
[0 0 1 tz]
[0 0 0 1]

Where (tx, ty, tz) are the translation factors along the x, y, and z axes, respectively.

A 3D rotation matrix can be represented as:

[cos(θ) -sin(θ) 0 0]
[sin(θ) cos(θ) 0 0]
[0 0 1 0]
[0 0 0 1]

Where θ is the angle of rotation.

A 3D scaling matrix can be represented as:

[sx 0 0 0]
[0 sy 0 0]
[0 0 sz 0]
[0 0 0 1]

Where (sx, sy, sz) are the scaling factors along the x, y, and z axes, respectively.

The composition of 3D transformations is similar to the composition of 2D transformations, where multiple transformations can be combined into a single matrix to achieve the desired result. To apply a 3D transformation to a given point (x, y, z), the point is multiplied by the transformation matrix to obtain the transformed point.

•••

B.C.A study

Unit-4:Representing curves and surfaces

In computer graphics, curves and surfaces are typically represented using mathematical models such as NURBS (Non-Uniform Rational B-Splines), Bézier curves, and Bézier surfaces. These models allow for precise control over the shape of the curve or surface and make it easy to manipulate and animate them. The parameters of these models can be manipulated to create a wide variety of shapes and styles, which makes them a popular choice in computer graphics and animation. To use these models in a CGMA (computer graphics and media applications) pipeline, one typically needs to have a strong understanding of the mathematical principles behind them, as well as the tools and software packages available for working with them

Polygon meshes parametric

A polygon mesh is a type of surface representation in computer graphics and animation that is made up of a set of polygons. These polygons can be either triangular or quadrilateral, and are defined by a set of vertices. The vertices are the points in space that define the shape of the polygons.

Parametric representation of a polygon mesh refers to the use of mathematical equations to define the position of each vertex in the mesh. This allows for precise control over the shape and form of the mesh, as well as the ability to easily modify and animate the mesh over time.

Example: Let's consider a torus (donut-shaped object) as an example. A torus can be represented as a parametric polygon mesh by using a mathematical equation to define the position of each vertex in the mesh.

One common way to create a torus is to use a parametric equation that defines the position of each vertex based on its polar and azimuthal angles. The polar angle determines the radius of the torus, while the azimuthal angle determines its position around the circumference of the torus.

To create the polygon mesh, we would divide the surface of the torus into a set of small polygons, each defined by a set of vertices. The position of each vertex is then determined using the parametric equation.

The resulting polygon mesh will have a precise, controlled shape that can be easily modified and animated over time. This makes parametric polygon meshes a powerful tool for creating complex shapes and objects in computer graphics and animation

cubic curves

Cubic curves are a type of curve representation in computer graphics and animation that are defined by a set of cubic polynomials. Cubic curves are a type of spline curve, which means that they are defined by a set of control points and are smooth and continuous between those control points.

Cubic curves are often used in CGMA for tasks such as modeling and animating smooth curves, such as those found in characters' limbs or objects' shapes. Cubic curves are also used in motion graphics to create smooth, animated transitions between keyframes.

Example: Let's consider a simple animation of a bouncing ball as an example. To create the animation, we could use cubic curves to define the path of the ball as it moves through the air.

We would start by defining a set of keyframes, each representing the position and velocity of the ball at a specific point in time. We would then use cubic curves to smoothly interpolate between these keyframes, defining the path of the ball as it moves through the air.

The resulting animation would have a smooth, continuous path for the ball, with no jagged or abrupt changes in its trajectory. This makes cubic curves a powerful tool for creating smooth, natural-looking animations in CGMA

•••

B.C.A study

Unit- 5 : Introductory concept of Multimedia

Multimedia

The word **'Multimedia'** is a combination of two words, **'Multi' and 'Media'.** Multi means many and media means material through which something can be transmitted or send. Multimedia combined all the media elements like text and graphics to make the information more effective and attractive. Now I am going to write about its components.

Components of Multimedia

The various components of multimedia are **Text, Audio, Graphics, Video and Animation.** All these components work together to represent information in an effective and easy manner.

1)Text: Text is the most common medium of representing the information. In multimedia, text is mostly use for titles, headlines,menu etc. The most commonly used software for viewing text files are *Microsoft Word, Notepad, Word pad etc.* Mostly the text files are formatted with ,DOC, TXT etc extension.

2)Audio: In multimedia audio means related with recording, playing etc. Audio is an important components of multimedia because this component increase the understandability and improves the clarity of the concept. audio includes speech, music etc. The commonly used software for playing audio files are:
i) Quick Time
ii) Real player
iii) Windows Media Player

3)Graphics: Every multimedia presentation is based on graphics. The used of graphics in multimedia makes the concept more effective and presentable.the commonly used software for viewing graphics are *windows Picture, Internet Explorer etc.* The commonly used graphics editing software is Adobe

Photoshop through which graphics can be edited easily and can be make effective and attractive.

4)Video: Video means moving pictures with sound. It is the best way to communicate with each other. In multimedia it is used to makes the information more presentable and it saves a large amount of time. The commonly used software for viewing videos are:
i) Quick Time
ii) Window Media Player
iii) Real Player

5)Animation: In computer animation is used to make changes to the images so that the sequence of the images appears to be moving pictures. An animated sequence shows a number of frames per second to produce an effect of motion in the user's eye. Some of the commonly used software for viewing animation are:
i) Internet Explorer
ii) Windows Pictures
iii) Fax Viewer

These are the components of Multimedia. Now I am going to talk about its applications.

Application of multimedia

Nowadays the application of Multimedia are observed in various fields such as Education, Entertainment,Business and so on. To communicate the message in the form of picture, sound, video, animation is the primary role of multimedia. Some of the application of multimedia are as follows:

1)Multimedia in Education: Multimedia is becoming popular in the field of education. It is commonly used to prepare study material for the students and also provide them proper understanding of different subjects.Nowadays Edutainment, a combination of Education and Entertainment has become very popular. This system provides learning as well as provides entertainment to the user.

2)Multimedia in Entertainment: Computer graphics techniques are now commonly use in making movies and games. this increase the growth of multimedia.

i)Movies: Multimedia used in movies gives a special audio and video effect. Today multimedia has totally changed the art of making movies in the world. Difficult effect, action are only possible through multimedia.

ii)Games: Multimedia used in games by using computer graphics, animation, videos have changed the gaming experience. Presently, games provides fast action, 3-D effects and high quality sound effects which is only possible through multimedia.

3)Multimedia in Business: Today multimedia is used in every aspect of business. These are some of the applications:

i)Videoconferencing: This system enables to communicate using audio and video between two different locations through their computers.When the information is sent across the world, this technology provides cost benefits to the business which saves their time, energy and money.

ii)Marketing and advertisement: Nowadays different advertisement and marketing ideas about any product on television and internet is possible with multimedia.

CD-ROM and Multimedia highway

CD-ROM (compact disc read-only memory, has become the most cost-effective distribution medium for multimedia projects: a CD-ROM disc can be mass-produced for pennies and can contain up to 80 minutes of full-screen video or sound. Or it can come rain unique mixes of images, sound, text, video and animations controlled by an authoring system to provide ultimates user interaction.

Discs can be stamped out of poly-carbonate plastic as fast as cookies on a baker's production line and just as cheaply. Virtually all personal computers sold today include a least a CD-ROM player, and the software that drives these computers is commonly available on a CD-ROM disc applications that required inserting as many as 16 or more floppy disk one after another are now installed from a CD-ROM without muss or fuss

Many systems now come with a DVD-ROM player, Multilayered Digital Versatile Disk (DVD) technology increases the capacity and multimedia capability or current optical technology to 18 GB. CD and DVD burners are used for reading discs and for making them, too, in audio, video, and data formats. DVD authoring and integration software allows the creation of interactive front-end menus for films and games.

In the very long term, however, CD-ROM and DVD discs are but interim memory technologies that will be replaced by new devices that do not require moving parts. As the data highway described below becomes more and more pervasive and users become better "connected", copper wire, glass Fiber, and radio/cellular technologies may prevail as the most common delivery means for interactive multimedia files, served across the broadband internet or from dedicated computer farms and storage facilities.

The Multimedia Highway:

Now, that telecommunications networks are global, and when information provides and content owners determines the worth of theirs products and how to charge money for them, information elements will ultimately link up online's as distributed resources on a data highway (actually more like a toll road). Where you will pay to acquire and use multimedia-based information.

Curiously, the actual glass Fiber cables that makes up much of the physical backbone of the data highway are, in many cases. Owned by railroad and pipelines companies who simply buried the cables on existing rights of way where no special permits and environmental reports are necessary. One railroad in the United States invested more than a million dollars in a special cable laying trenching car; in the United Kingdom, there is talk of placing a fiber-optic cables backbone along the decaying 19th century canal and barge system. Bandwidth on these lines is leased to other, so competing retailers such as AT&T, MCI, and Sprint may even share the same cable. Full-text content from books and magazines is accessible by modem and electronic link; features movies are played at home; real-time new reports from anywhere on earth are available; lectures from participating universities are monitored for education credits; street maps of any city are view-able with recommendations for restaurants, in any language-and online travelogues include testimonials and video tracks. This is not science fiction; it is happing now. For each of these interfaces or gateways to information is a Multimedia projects just waiting to be developed.

Definition of Animation

“One of the most exciting forms of pictoral presentation is animation. Animation refers to a simulated motion picture depictingmovementof drawn (or simulated) objects. The main features of this definition are as follows: (1) picture – an animation is a kind of pictorial representation; (2) motion – an animation depicts apparent movement; and (3) simulated – an animation consists of objects that are artificially created through drawing or some other simulation method.” (Mayer 2002:88)

“The potential educational benefits of animation arise from its capacity to portray temporal change directly and explicitly” (Lowe and Schnotz, ?)

Usages and function of multimedia animation in education

3.1 Typical usages

- To attract attention
- To inform about the state of process (e.g. progress bars that show the percentage of program loading)
- Demonstrations (e.g. show how a volcano may interrupt by moving tectonic plates)
- Interactive simulation (e.g. have a learner fuel and point a rocket and then show its flight path)

3.2 Pedagogical function

- Motivation, get students interested in some phenomenon and to explore it.
- Representation, help to support mental representation
- Organization
- Interpretation, provoke cognitive conflicts that make the students think.

Types of Animation Techniques

1. Traditional animation
2. 2D animation
3. 3D animation
4. Typography Animation
5. Clay animation
6. Sand Animation
7. Flip book Animation
8. Stop-motion animation,

Traditional animation or Classical 2D animation :

Traditional animation involved animators drawing by hand for each and every frame. If you love the feel of pencils on a paper, then the traditional approach is very fascinating. Traditional animation is creating the drawings one by one on the frame. 2D animation involves creating numerous drawings then feeding into a plastic cells, hand painting them and create the animated sequence on a painted background image.
Traditional Aniamtion Movies : Snow White and the Seven Dwarfs, Peter Pan, and Sleeping Beauty, Aladdin

Digital 2D animation

Creating animations in the 2 dimensional space with the help of digital technologies is known as digital 2d animation. You don't need to create digital models, you just need to draw the frames. Create 100s of drawing and animating them to show some kind of movement is technically known as digital 2d animation. Using Adobe flash, animators can limit the number of drawings used, which makes them easier to create digital 2d animation. Small variations like changing the color or frame rate can be changed almost instantly, thus making it easier for the animators to work on.

: Digital 3D animation

If you are interested in making the unreal characters into a realistic one, then it's Digital 3d animation. Digital 3d animation characters are much faster to create and they are quite popular in the movie making industry. Using a computer software 3d animated images are used to create many short films, full length movies and even tv commercials and a career in digital 3d animation is highly rewarding. Comparing to 2D animation and the traditional approach, 3d animation models are highly realistic.

Stop-motion animation

Have you ever wondered if a piece of stone can walk or talk, well anything is possible in animation? Using frame by frame animation, physical static objects are moved around and during the post production it is shown in a fluid movement. Stop motion animation has been around ever since the evolution of puppets. There were many movies created using the stop motion method, some of the finest examples are “Fun in a bakery shop” created in 1902.Edwin Porter directed “The Teddy Bears,” which was one of the earliest stop-motion animation films. The movie is a short sequence of playing teddy bears, just over a minute in length, which took over 50 hours to animate.

Mechanical Animation

Instead of robotics, machines can be animated by using the mechanical animation technique. Instead of creating the original machine, creating these mechanical animations, allows the animator to understand how the machine works. Explaining the functionality of these machines is quite easy through this kind of animation technique.

6 : Audio-Animatronics and Autonomatronics

Walt disneyimagineering created the trademark audio animatronics which is fitted in its disney theme parks. Otto is a robot which can easily sense a person in a room, converse with them and can also tell if they are happy. Autonomatronics technology is different from Audio-Animatronics technology. Audio-Animatronics technology repeats a pre-programmed show over and over again. Autonomatronics technology is driven by sophisticated cameras and sensors giving Otto the ability to make choices about what to say and do.

7: Chuckimation

Chuckimation is one of the popular animation techniques created by “Action League Now!” creators. It’s a combination of stop frame animation and live shots, where characters are dropped into a particular frame. It has some similarities to the famous puppet shows.

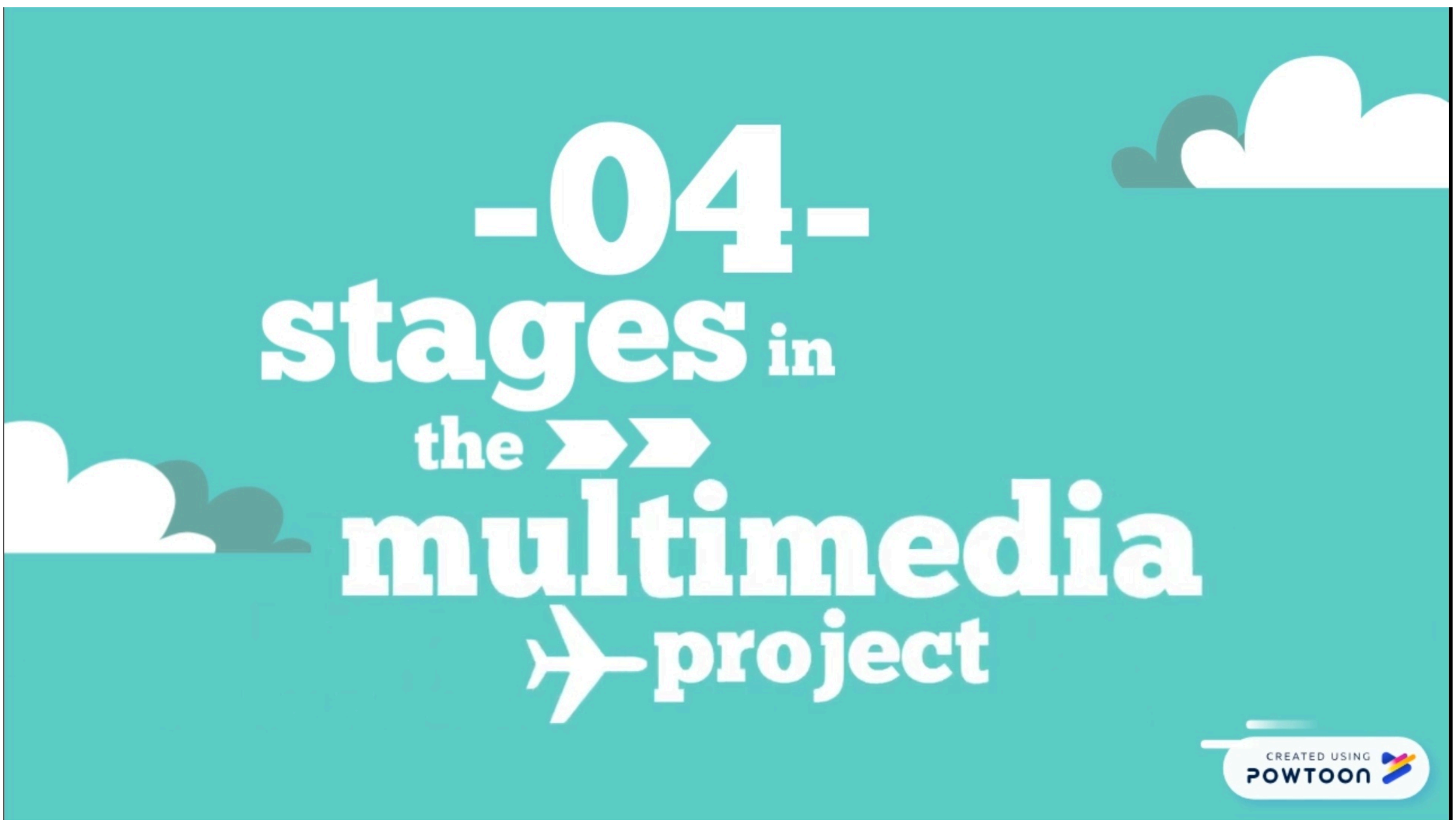
: Puppetry Animation

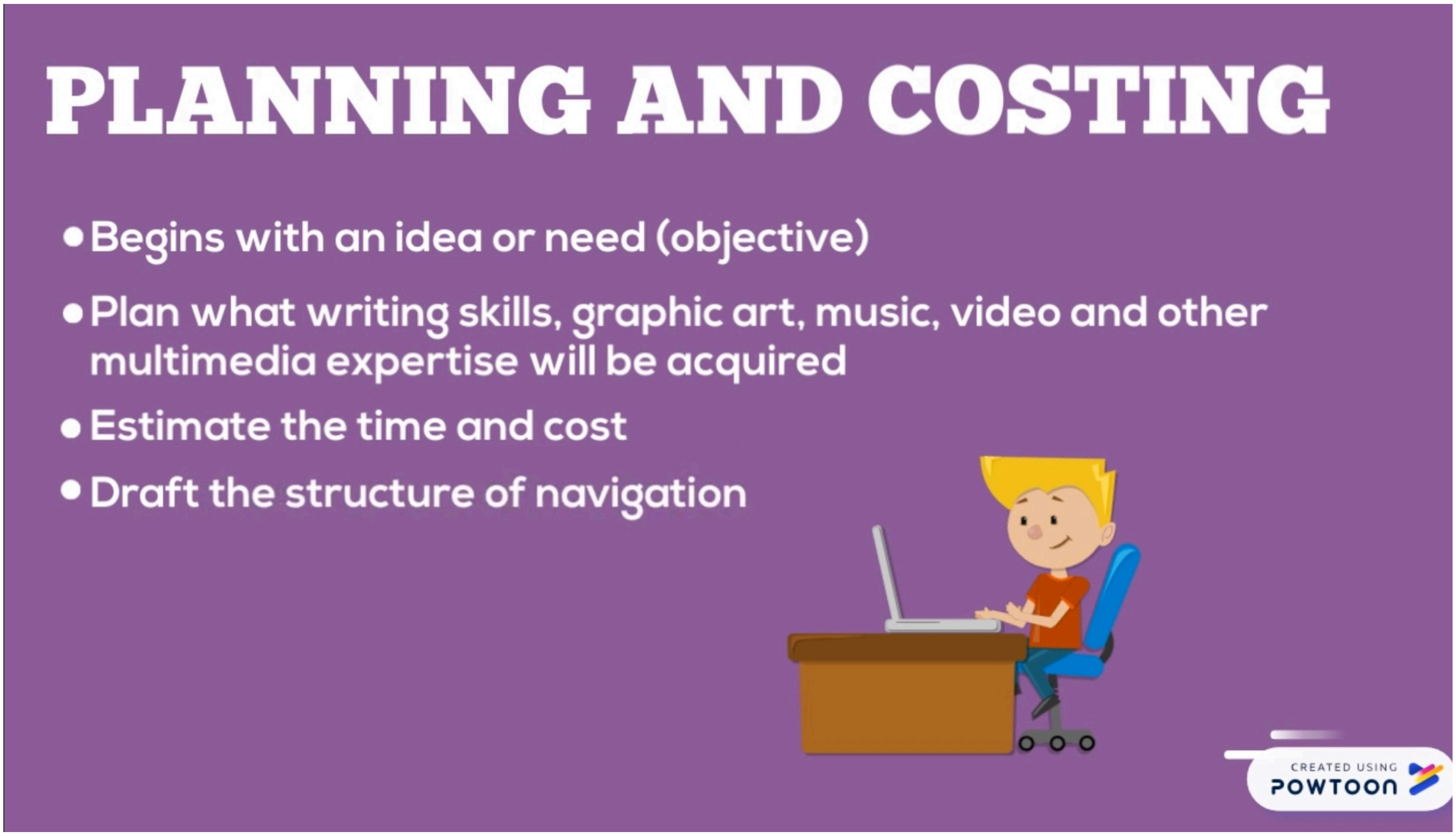
Puppetry animation is created using the life like puppets instead of objects. The film 'The Humpty Dumpty Circus' (1908) created by J. Stuart Blackton and Albert smith receives credit as the first stop-motion animation film that features puppets. Nowadays puppet animation is most commonly used in children's cartoons and films. An example of puppet animation used in cinema is in the film King Kong (1933).The Nightmare before Christmas (1993) is an American stop motion musical fantasy horror film directed by Henry Selick. It used 227 puppets to represent the characters in the film and also 400 heads were used to allow the expression for every possible emotion.

B.C.A study

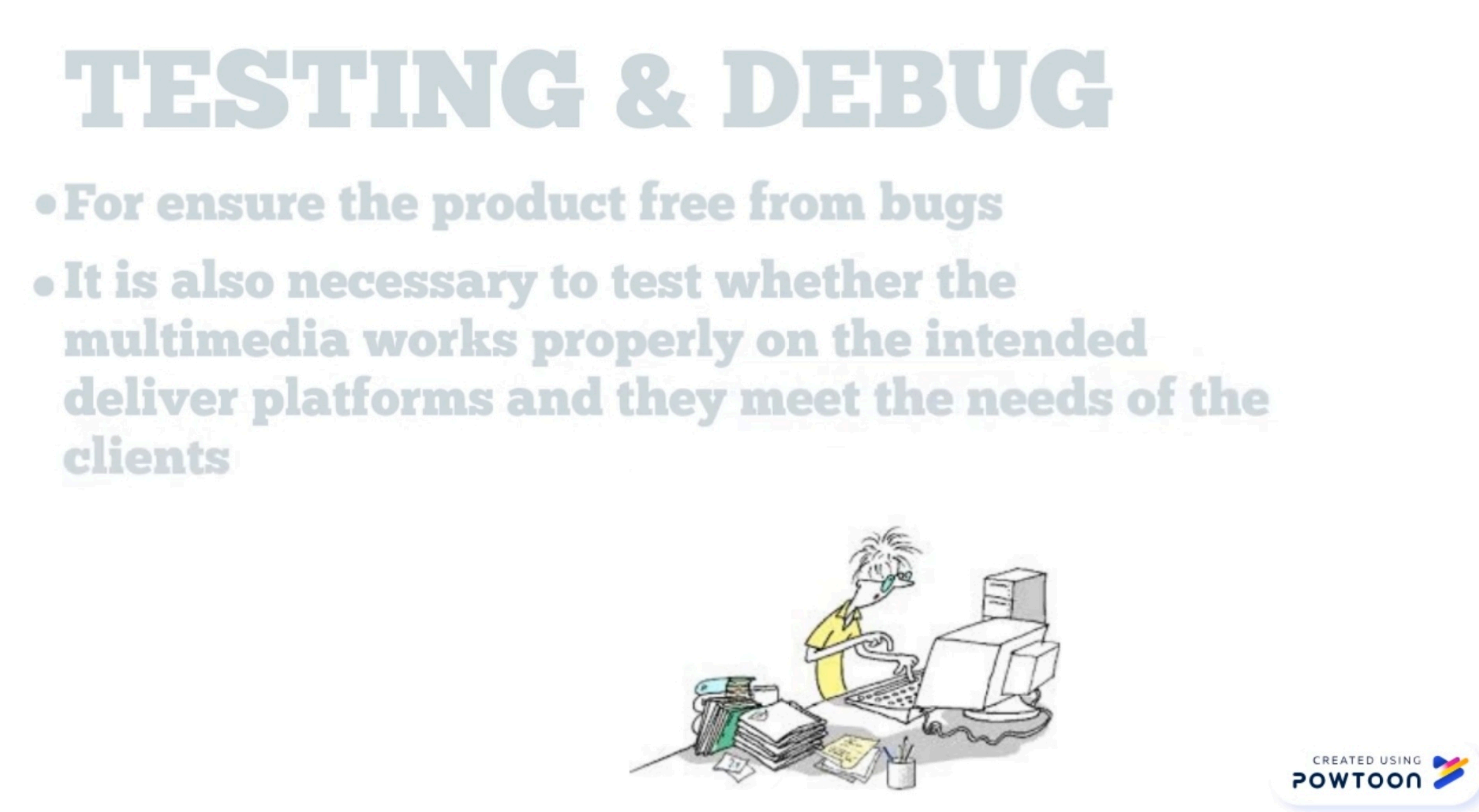
Unit -6 :multimedia components

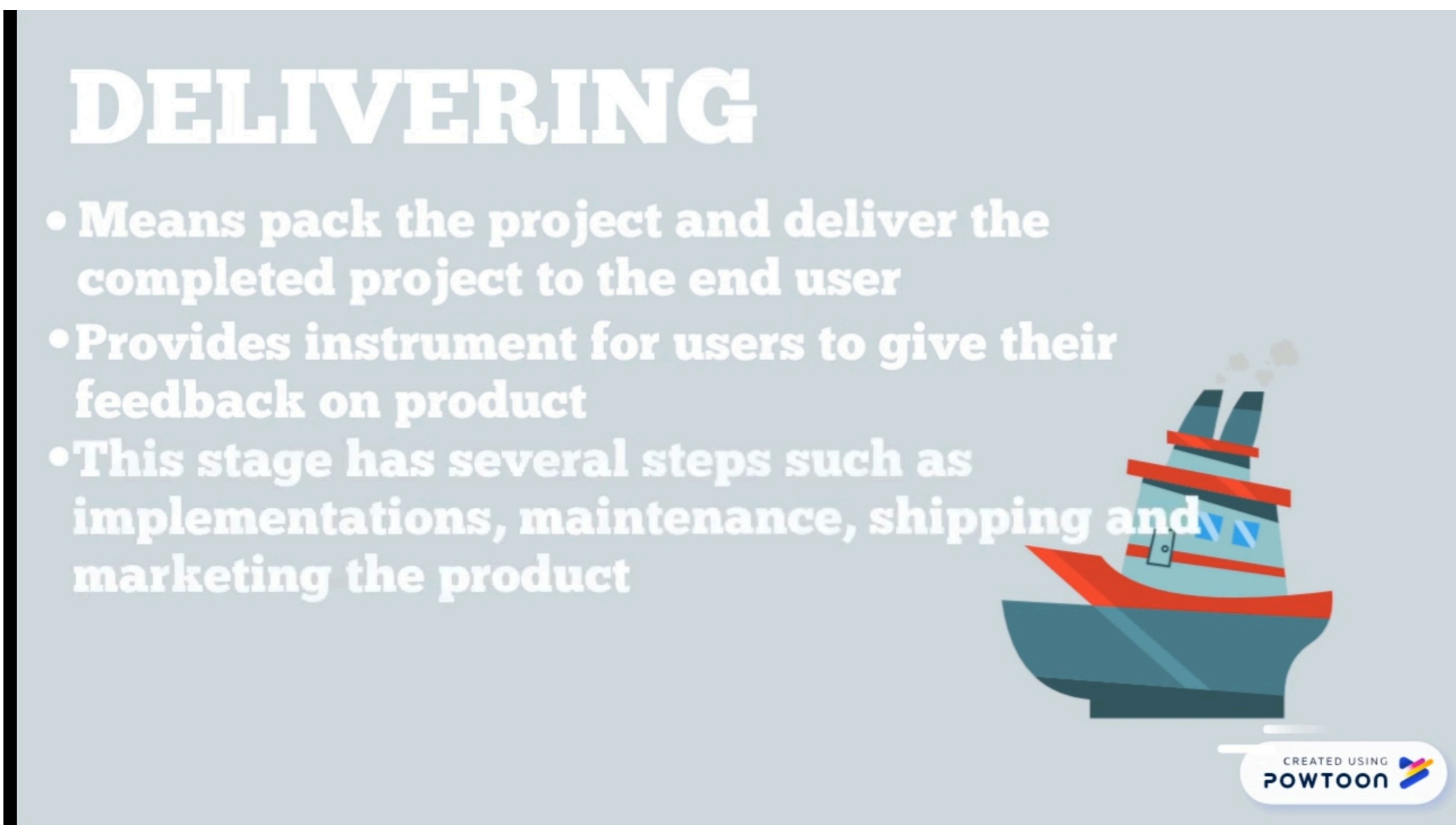
Introduction to making multimedia




PLANNING AND COSTING
• Begins with an idea or need (objective)
• Plan what writing skills, graphic art, music, video and other multimedia expertise will be acquired
• Estimate the time and cost
• Draft the structure of navigation
CREATED USING POWTOON


DESIGNING AND PRODUCING
• Storyboarding the project
• Acquiring the multimedia resources
• Design the navigational structure and interface
• Execute each of the planned tasks and create a finished product


TESTING & DEBUG
• For ensure the product free from bugs
• It is also necessary to test whether the multimedia works properly on the intended deliver platforms and they meet the needs of the clients
CREATED USING
POWTOON


DELIVERING
• Means pack the project and deliver the completed project to the end user
• Provides instrument for users to give their feedback on product
• This stage has several steps such as implementations, maintenance, shipping and marketing the product
CREATED USING
POWTOON

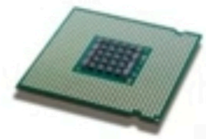

HARDWARE & SOFTWARE
REQUIRED FOR MULTIMEDIA


BASIC HARDWARE REQUIREMENT
The components of a Multimedia Computer System can be divided into the five categories:
System Devices
Memory & Storage Devices
Input Devices
Output Devices
Communication Devices

SYSTEM DEVICES

System devices in a multimedia computer system includes the following major components:

- **Motherboard**
  It connects all the pars of the computer including CPU, memory, hard drives, and other ports and expansion cards and all. It is the "backbone" of the PC



- **Microprocessor**
  It is the "engine" of the computer and is some way responsible for every single thing the computer does.

- **Graphics Processing Unit**
  A single-chip processor to manage and boost the performance of video and graphics.

MEMORY & STORAGE DEVICES

Memory and Storage devices are used to store programmes or data on a temporary or permanent basis for use in computer. This Memory devices are includes:

- **Random Access Memory**
  Also called the primary memory of the Computer and is responsible for containing the operating system, data and the application programs

- **Read-Only Memory**
  Read-only memory is non-volatile like RAM. When you turn off the power to a ROM chip does not lose its memory. ROM is typically used in computers to hold fixed applications the small BIOS program that initially boots up the computer and it is used in printers to hold built-in fonts.

MEMORY & STORAGE DEVICES

Some of the important Storage devices are:

- **Hard Disc**
  Most common mass-storage device used on computers, and for making multimedia operations. It stores and provides access to large amounts of data on electro magnetically charged surface.
- **DVD**
  A digital optical disc storage format that can store any kind of digital data. It offers higher storage capacity than compact discs while having the same dimensions.
- **Compact Disc (CD)**
  An optical medium that can store data pertaining to audio, video, text and other information in digital form. These are Read-Only devices, however newer technology allow users to record as well.
- **Compact Disc – Read Only Memory**
  Known as CD-ROM can store data in the form of text, graphics and sound. There are a few types of CD ROMs such as CD-R or CD Recordable and CD-Rewritable or CD-RW.

INPUT DEVICES

An input device is any hardware device that sends data to a computer, allowing you to interact with and control the computer. Some of the Important input Devices are:

- **Keyboards**
  The most common input device used to interact with the computer. It is primarily used for send text input and some command for your computer.
- **Pointing Device**
  Pointing device is an input interface that allows user to control and provide data to the computer using physical gestures by moving a hand-held mouse or similar devices.
- **Composite devices**
  Input devices, such as buttons and joysticks, can be combined on a single physical device that used as a composite device. Many gaming devices have controllers like this.
- **Scanners**
  It captures images for computer editing and display. Scanners come in hand-held, feed-in, and flatbed types and for scanning black-and-white only, or color.
- **Audio input devices**
  Microphone is another input device that can capture audio. Through MIDI the musical instruments can also be used as input devices.

OUTPUT DEVICES

Output devices receive the processing from the computer and presents or displays it. Some of the important output devices are:

- **Printer**

  It receives signal from computer transfers the information to paper. The colour printer help to clarify concepts, improve understanding and retention of information as well as helps to organize complex data.
- **Monitor**

  Display device like a television set which displays information in pictorial form. it is measured diagonally from two opposing corners of the picture tube.
- **Amplifier**

  An electronic device that increases the power of a signal. Amplifiers are used in audio equipment. They are also called power amplifiers. Speakers with built-in amplifiers have become an integral part of the computers today and are important for any multimedia project.

CONFIGURATION OF A MULTIMEDIA COMPUTER

A GOOD MULTIMEDIA SYSTEM SHOULD HAVE

- **Processor: a Pentium 3.0 GHz onwards or similar**
- **RAM: 4GB or more**
- **Hard Disk Drive: 1TB or more**
- **Monitor Size: 19 Inch or more**
- **GPU: 1GB VRAM PCI express card**
- **DVD Writer: Dual Layer**
- **Sound Card: 5:1 or 7:1 bit**
- **high wattage sub-woofer speakers,**
- **Standard Multimedia keyboard,**
- **3 button mouse and a touch panel.**

You can also add a scanner, printer, digital camcorder and a video-capture card. There is no set rule to define the exact hardware combination of a good multimedia computer.

BASIC SOFTWARE REQUIREMENT

Basic tool set for building multimedia project can be divided into following categories:

- Painting and drawing tools
- 3-D Modelling and animation tools
- Image editing tools
- Sound editing tools
- Animation video and Digital movie tools

SOFTWARE FOR MULTIMEDIA

Word Processor
Used for the production (including composition, editing, formatting, and possibly printing) of any sort of printable material. Some of the popular word processing applications are: Microsoft Office Word, WordPerfect, OpenOffice, AbiWord etc.

Graphic Software Packages
Allows you to create graphic figures and other things.
Graphic Packages Applications include: Photoshop, Illustrator, Paint Shop Pro, MS paint

Audio Software Packages
Used to edit and modify audio clips. Some examples of Audio Software Packages are: Audacity, WavePad Sound Editor, GoldWave, Power Sound Editor, Wavosaur etc.

PAINTING & DRAWING TOOLS

Video Software Packages
Captures and edits the video from a source such as a camcorder or VCR and then "capture" it in the computer. Manipulates the source video and combine it with other elements to create a finished video. Editing software is non-linear, meaning you can take a single scene out of the original video and move it into any location onto the new video project. Video Software packages examples include: Adobe Premiere, iMovie, Sony Vegas Movie Studio, Windows Movie Maker etc.

Animation Software
Allows graphics to be created, manipulated and edited into an animation where characters are brought to "life". Popular 2D Animation Software include Toon Boom Studio 4, Animationish, Flip Boom, Flash, After Effects, CreaToon etc. whereas 3D Studio Max, Maya, Lightwave, Blender, Anim8tor, Swift 3D etc are used for 3D animation.

SOFTWARE FOR MULTIMEDIA

CD/DVD-Burner Software
These programmes allow you to record/write text, video, sound and graphics into CD/DVD thus storing information into the device. Examples of CD/DVD burner software: NCH Express Burn, CDBurnerXP, Nero Burning ROM

Web Publishing Packages
These packages help in design and create projects and then publish the project onto the web. Examples of web publishing software: Blogger, Adobe dreamweaver, iWeb (Apple Macintosh), Muse

CHOOSING RIGHT SOFTWARE

While selecting the software for multimedia projects, one should consider the following features

Usability: Should have a capability to deal with a variety of text, images video and sound formats with precision and ease.

Animations: Should have wide ranging capabilities in terms of interactive simulations, media support, animated buttons, illustrations, maps, etc.

Smoothness: Should have anti-aliasing feature, meaning that all letter and image edges are smooth.

Integration: Should have integration capabilities with a wide range of software used for different jobs like Real, ActiveX, Shockwave, Flash, QuickTime, Photoshop and other applications .

Delivery: Should be able to develop one piece of content for delivery on different media types.

User friendliness: Should be the easiest, most versatile, and have the most pre-built models

Clientele : Should have applications for instructional designers, subject matter experts, training developers and others.